BACKGROUND ART

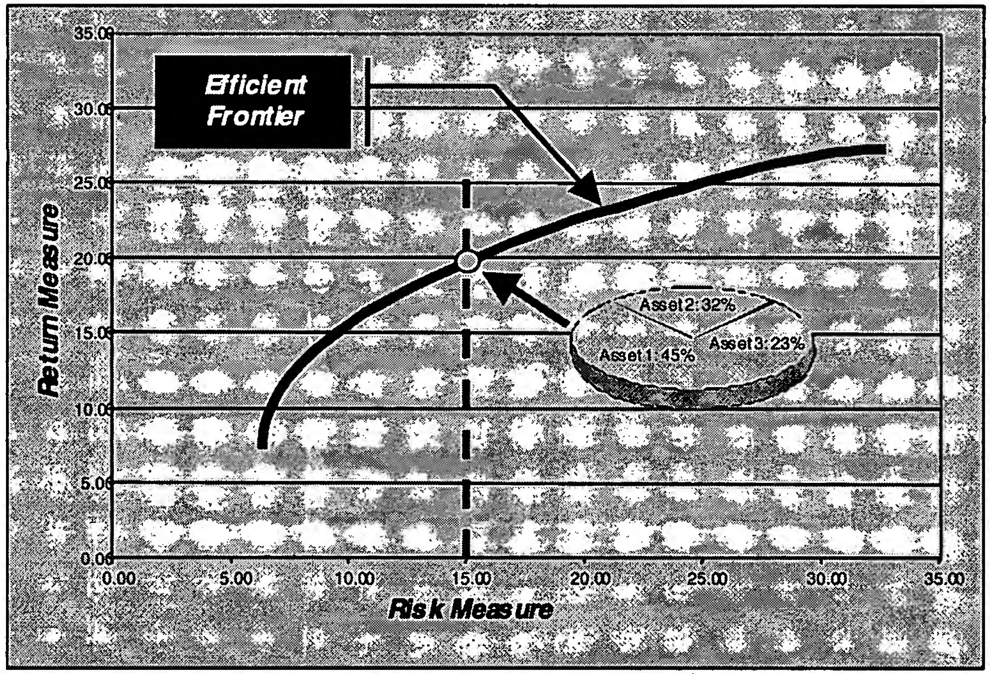
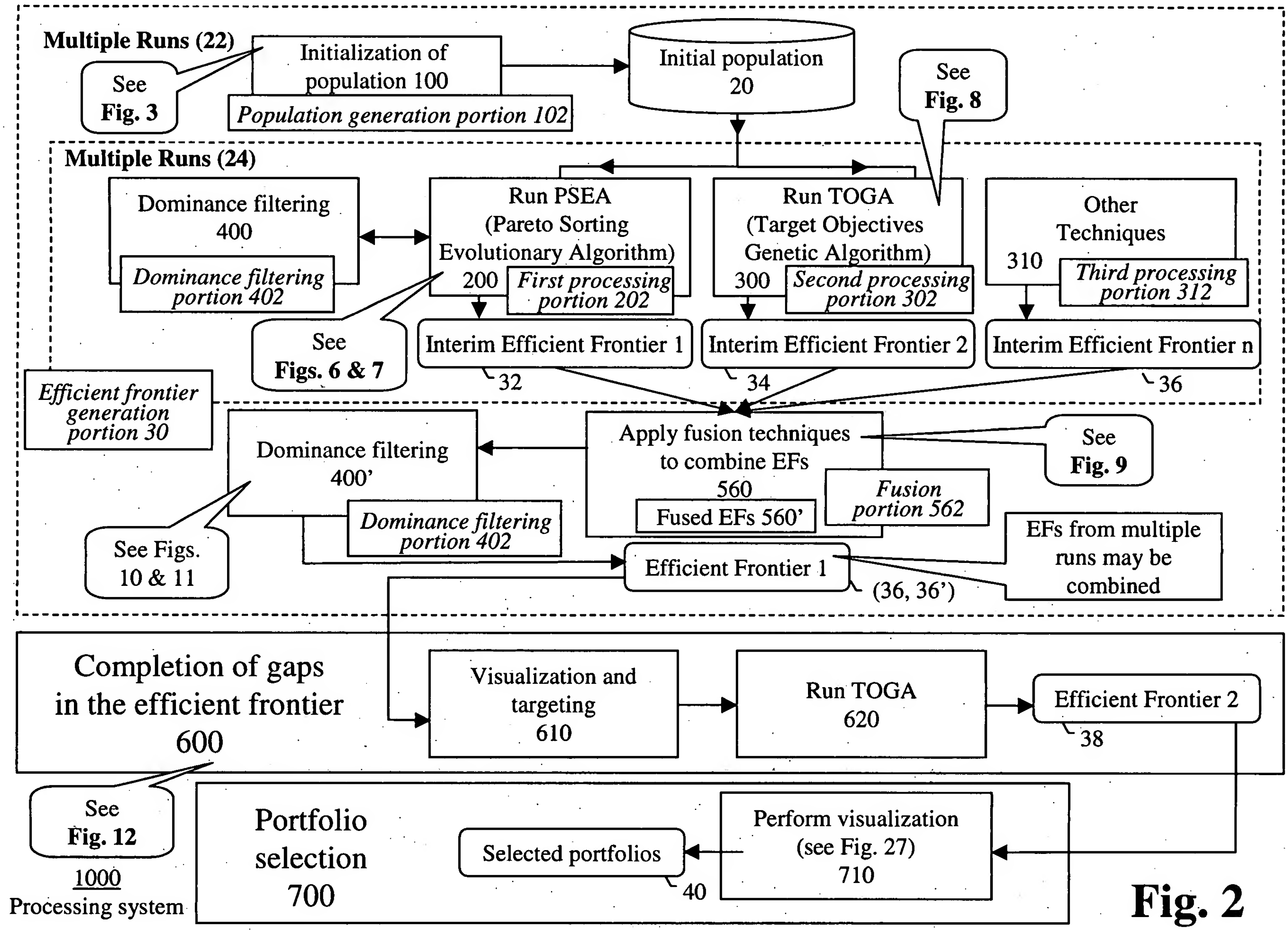
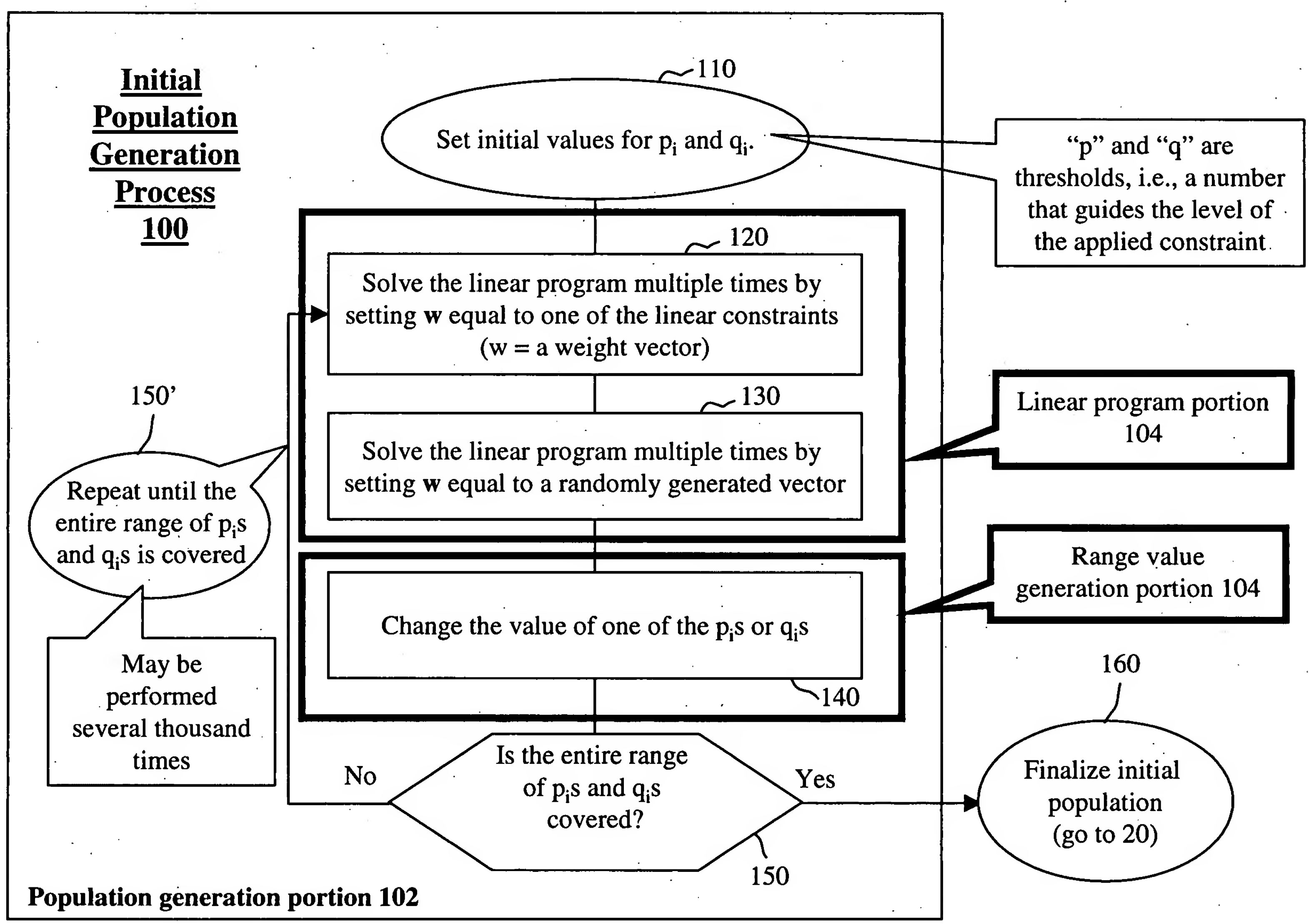


Fig. 1



Multiple Runs (22)
See Fig. 3
Initialization of population 100
Population generation portion 102
Initial population 20
See Fig. 8
Multiple Runs (24)
Dominance filtering 400
Dominance filtering portion 402
Run PSEA (Pareto Sorting Evolutionary Algorithm)
200
First processing portion 202
Run TOGA (Target Objectives Genetic Algorithm)
300
Second processing portion 302
Other Techniques
310
Third processing portion 312
See Figs. 6 & 7
Interim Efficient Frontier 1
32
Interim Efficient Frontier 2
34
Interim Efficient Frontier n
36
Efficient frontier generation portion 30
Dominance filtering 400'
Dominance filtering portion 402
See Figs. 10 & 11
Apply fusion techniques to combine EFs 560
Fused EFs 560'
Fusion portion 562
See Fig. 9
EFs from multiple runs may be combined
Efficient Frontier 1
(36, 36')
Completion of gaps in the efficient frontier 600
Visualization and targeting 610
Run TOGA 620
Efficient Frontier 2
38
See Fig. 12
1000
Processing system
Portfolio selection 700
Selected portfolios
40
Perform visualization (see Fig. 27) 710

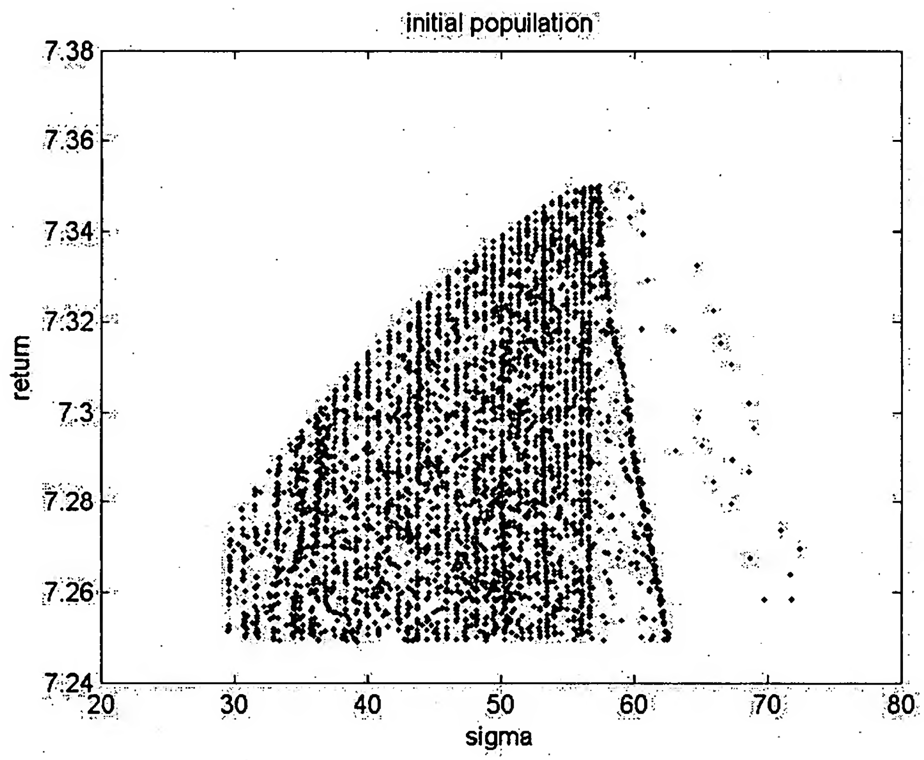
Fig. 2



Initial Population Generation Process 100
110
Set initial values for p_i and q_i.
"p" and "q" are thresholds, i.e., a number that guides the level of the applied constraint
120
Solve the linear program multiple times by setting **w** equal to one of the linear constraints (w = a weight vector)
130
Solve the linear program multiple times by setting **w** equal to a randomly generated vector
Linear program portion 104
150'
Repeat until the entire range of p_is and q_is is covered
May be performed several thousand times
Change the value of one of the p_is or q_is
140
Range value generation portion 104
160
Is the entire range of p_is and q_is covered?
No
Yes
150
Finalize initial population (go to 20)
Population generation portion 102

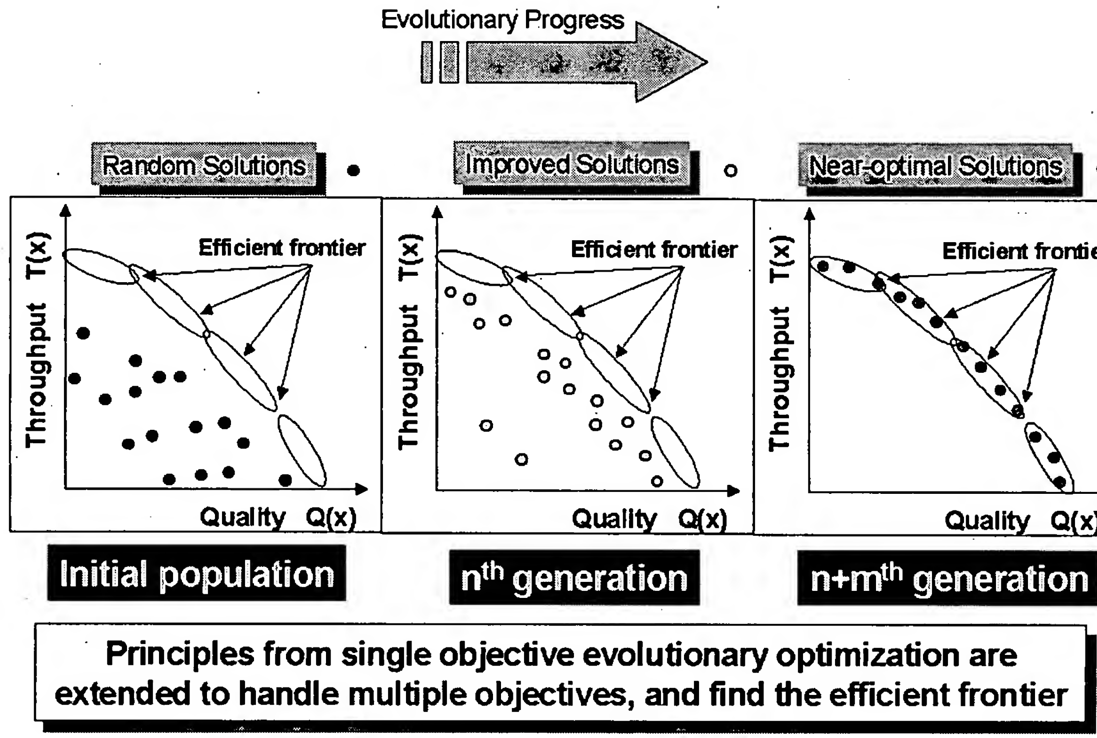
Fig. 3

Fig. 4



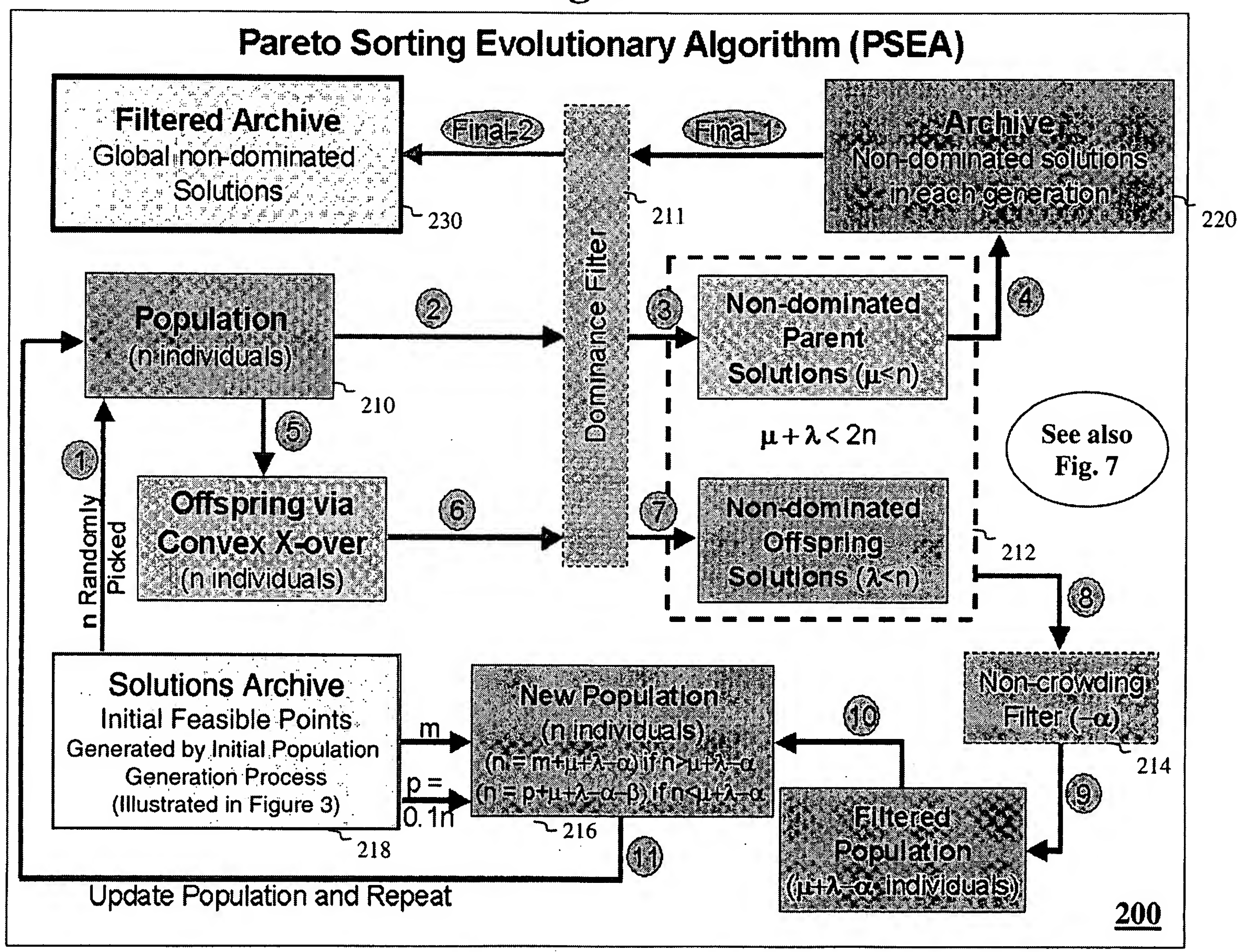
initial population
return
7.38
7.36
7.34
7.32
7.3
7.28
7.26
7.24
20
30
40
50
60
70
80
sigma

Fig. 5



Evolutionary Progress
Random Solutions
Improved Solutions
Near-optimal Solutions
Efficient frontier
Efficient frontier
Efficient frontier
Throughput T(x)
Throughput T(x)
Throughput T(x)
Quality Q(x)
Quality Q(x)
Quality Q(x)
Initial population
nth generation
n+mth generation
Principles from single objective evolutionary optimization are extended to handle multiple objectives, and find the efficient frontier

Fig. 6



Pareto Sorting Evolutionary Algorithm (PSEA)
Filtered Archive
Global non-dominated Solutions
230
Final-2
Dominance Filter
211
Final-1
Archive
Non-dominated solutions in each generation
220
Population
(n individuals)
210
2
3
Non-dominated Parent Solutions (μ<n)
4
5
1
n Randomly Picked
Offspring via Convex X-over
(n individuals)
6
7
μ+λ<2n
Non-dominated Offspring Solutions (λ<n)
212
See also Fig. 7
8
Solutions Archive
Initial Feasible Points
Generated by Initial Population Generation Process
(Illustrated in Figure 3)
218
m
p = 0.1n
New Population
(n individuals)
(n = m+μ+λ−α) if n>μ+λ−α
(n = p+μ+λ−α−β) if n<μ+λ−α
216
10
Non-crowding Filter (−α)
214
9
Filtered Population
(μ+λ−α individuals)
11
Update Population and Repeat
200

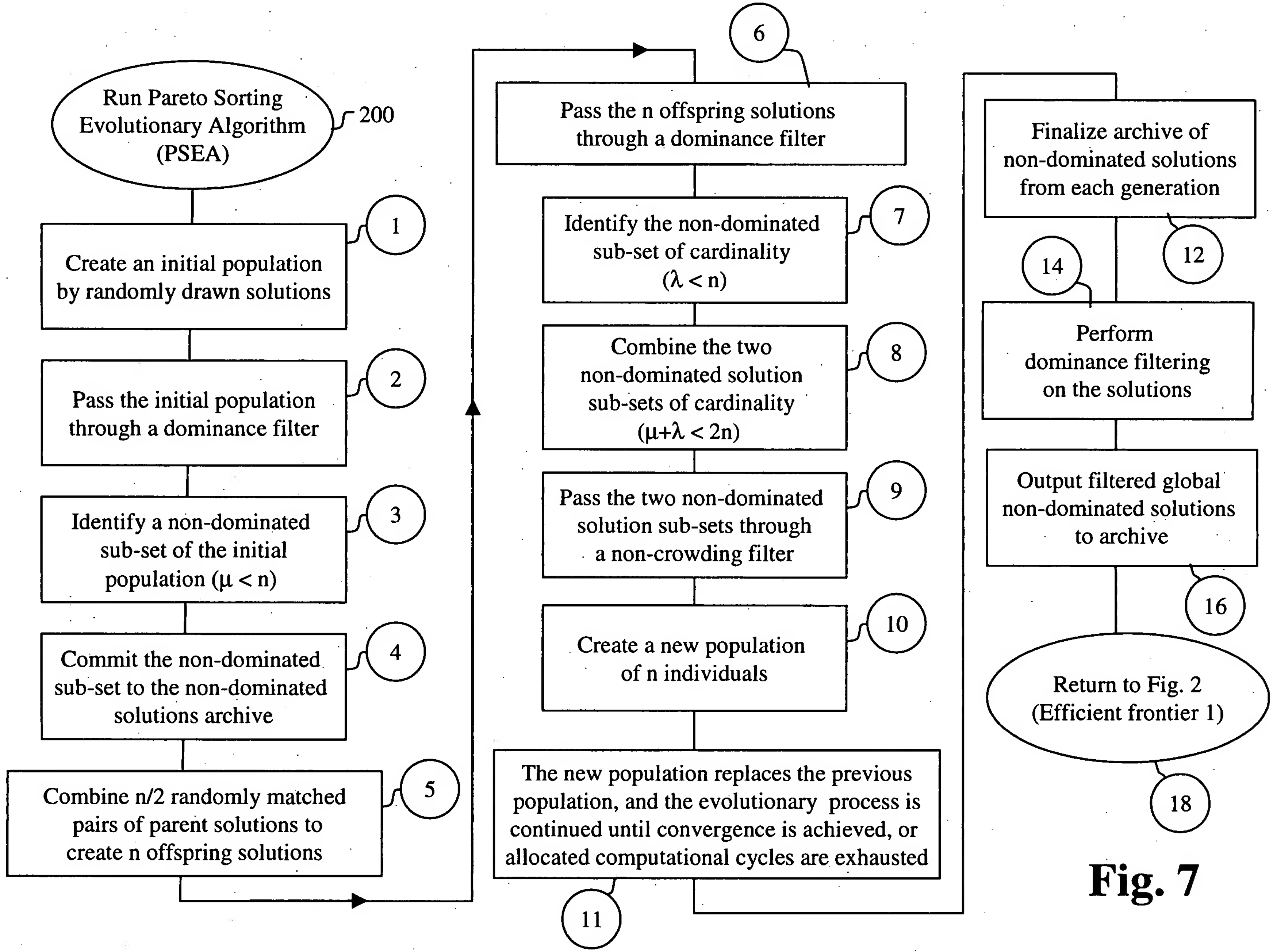
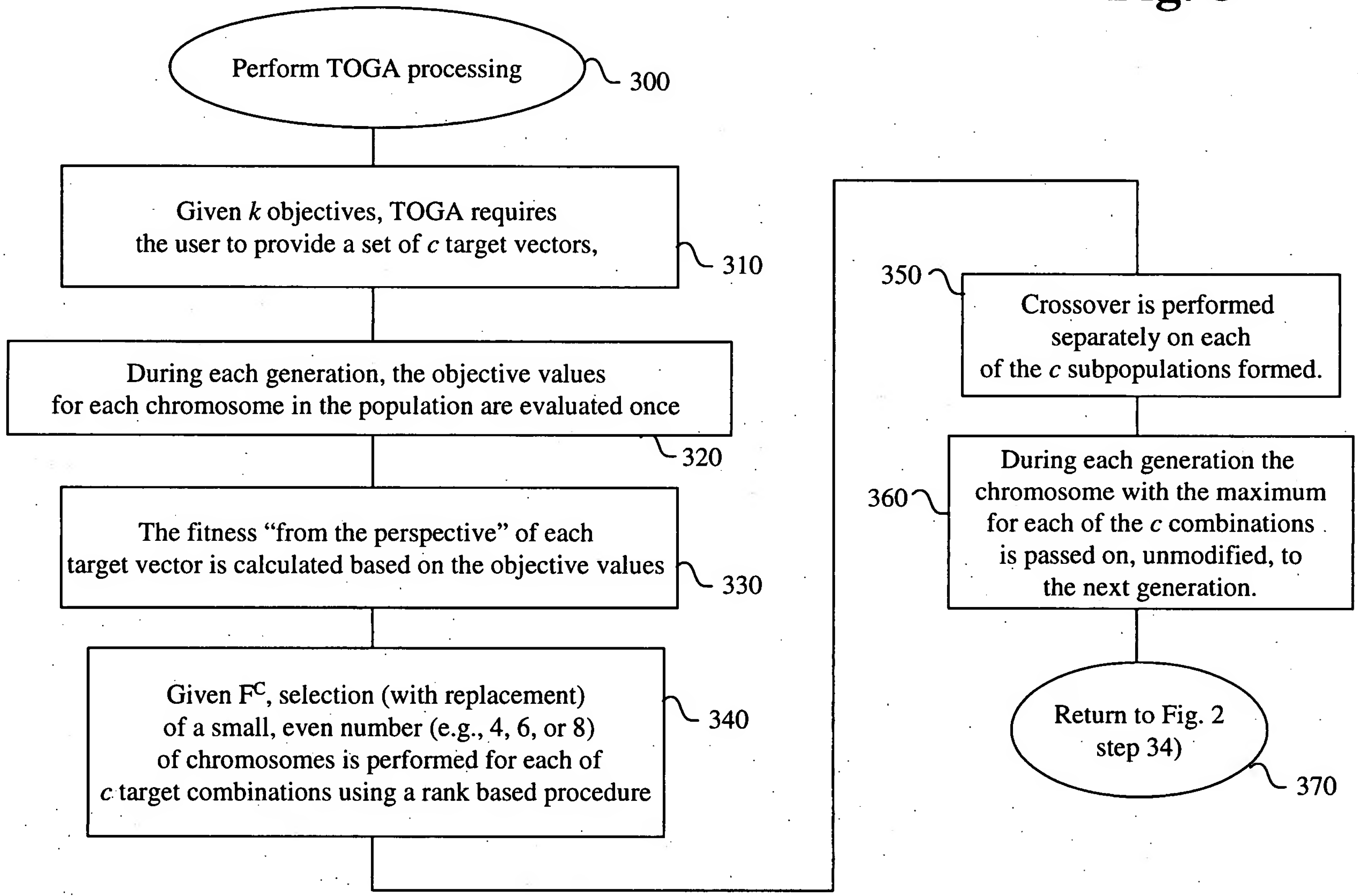
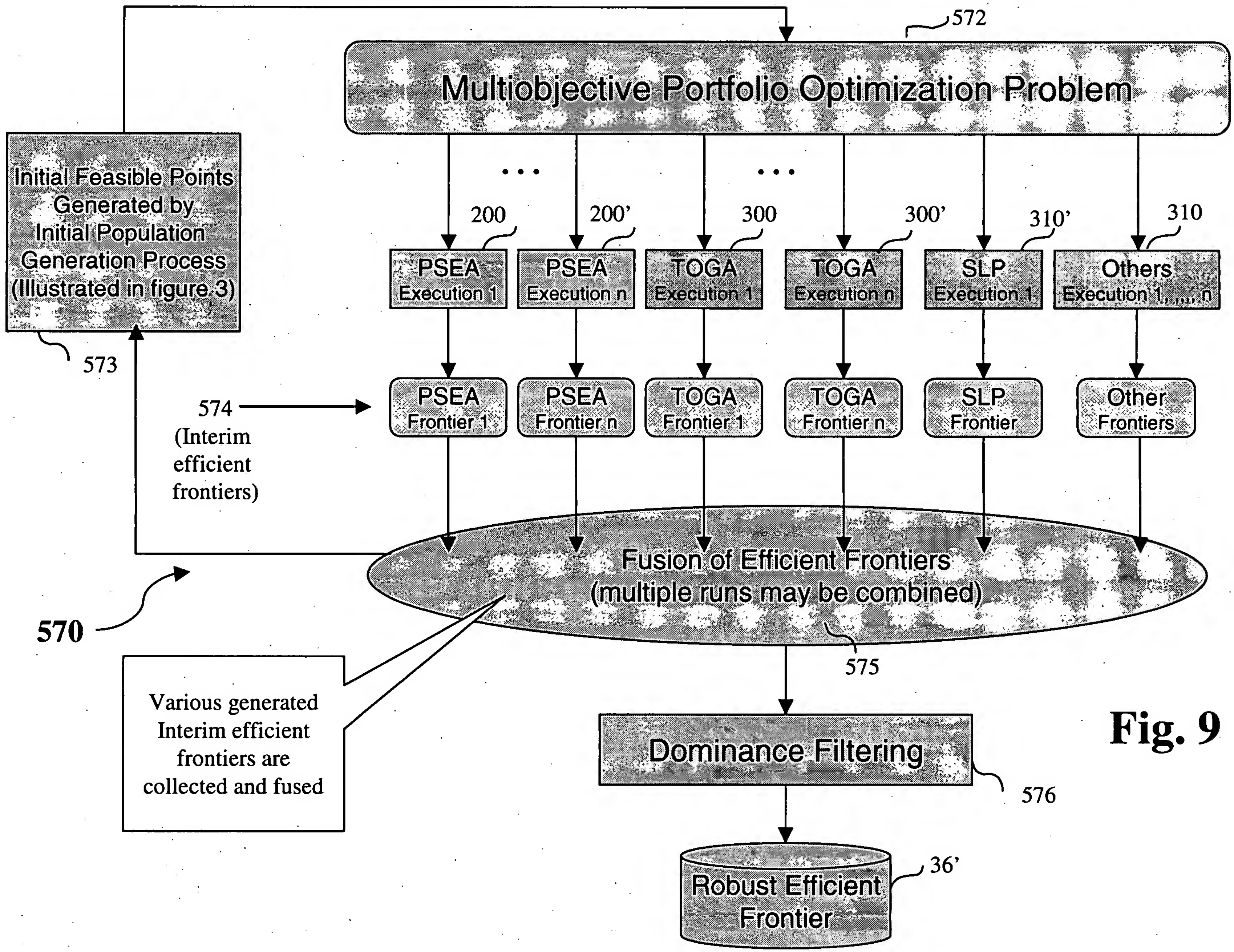

Run Pareto Sorting Evolutionary Algorithm (PSEA)
200
Create an initial population by randomly drawn solutions
1
Pass the initial population through a dominance filter
2
Identify a non-dominated sub-set of the initial population (μ < n)
3
Commit the non-dominated sub-set to the non-dominated solutions archive
4
Combine n/2 randomly matched pairs of parent solutions to create n offspring solutions
5
Pass the n offspring solutions through a dominance filter
6
Identify the non-dominated sub-set of cardinality (λ < n)
7
Combine the two non-dominated solution sub-sets of cardinality (μ+λ < 2n)
8
Pass the two non-dominated solution sub-sets through a non-crowding filter
9
Create a new population of n individuals
10
The new population replaces the previous population, and the evolutionary process is continued until convergence is achieved, or allocated computational cycles are exhausted
11
Finalize archive of non-dominated solutions from each generation
12
Perform dominance filtering on the solutions
14
Output filtered global non-dominated solutions to archive
16
Return to Fig. 2 (Efficient frontier 1)
18

Fig. 7

Fig. 8



Perform TOGA processing
300
Given k objectives, TOGA requires the user to provide a set of c target vectors,
310
During each generation, the objective values for each chromosome in the population are evaluated once
320
The fitness "from the perspective" of each target vector is calculated based on the objective values
330
Given F^C, selection (with replacement) of a small, even number (e.g., 4, 6, or 8) of chromosomes is performed for each of c target combinations using a rank based procedure
340
350
Crossover is performed separately on each of the c subpopulations formed.
360
During each generation the chromosome with the maximum for each of the c combinations is passed on, unmodified, to the next generation.
Return to Fig. 2 step 34)
370



572
Multiobjective Portfolio Optimization Problem
Initial Feasible Points Generated by Initial Population Generation Process (Illustrated in figure 3)
573
200
200'
300
300'
310'
310
PSEA Execution 1
PSEA Execution n
TOGA Execution 1
TOGA Execution n
SLP Execution 1
Others Execution 1, ..., n
574
(Interim efficient frontiers)
PSEA Frontier 1
PSEA Frontier n
TOGA Frontier 1
TOGA Frontier n
SLP Frontier
Other Frontiers
Fusion of Efficient Frontiers
(multiple runs may be combined)
575
570
Various generated Interim efficient frontiers are collected and fused
Dominance Filtering
576
Robust Efficient Frontier
36'

Fig. 9

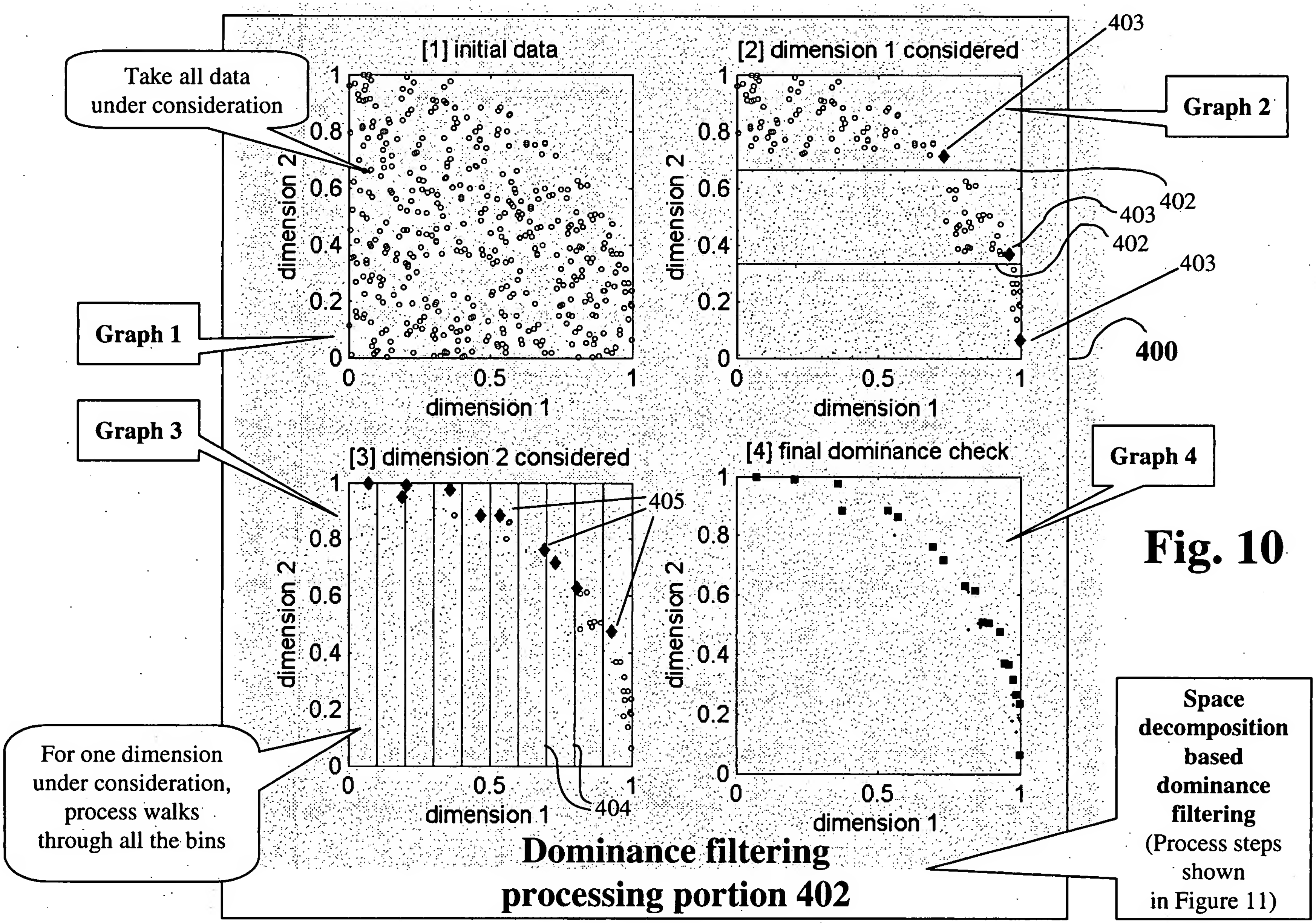

Take all data under consideration
Graph 1
Graph 2
Graph 3
Graph 4
[1] initial data
[2] dimension 1 considered
[3] dimension 2 considered
[4] final dominance check
dimension 1
dimension 2
0
0.2
0.4
0.6
0.8
1
0.5
403
402
405
404
400
For one dimension under consideration, process walks through all the bins
Dominance filtering processing portion 402
Space decomposition based dominance filtering (Process steps shown in Figure 11)

Fig. 10

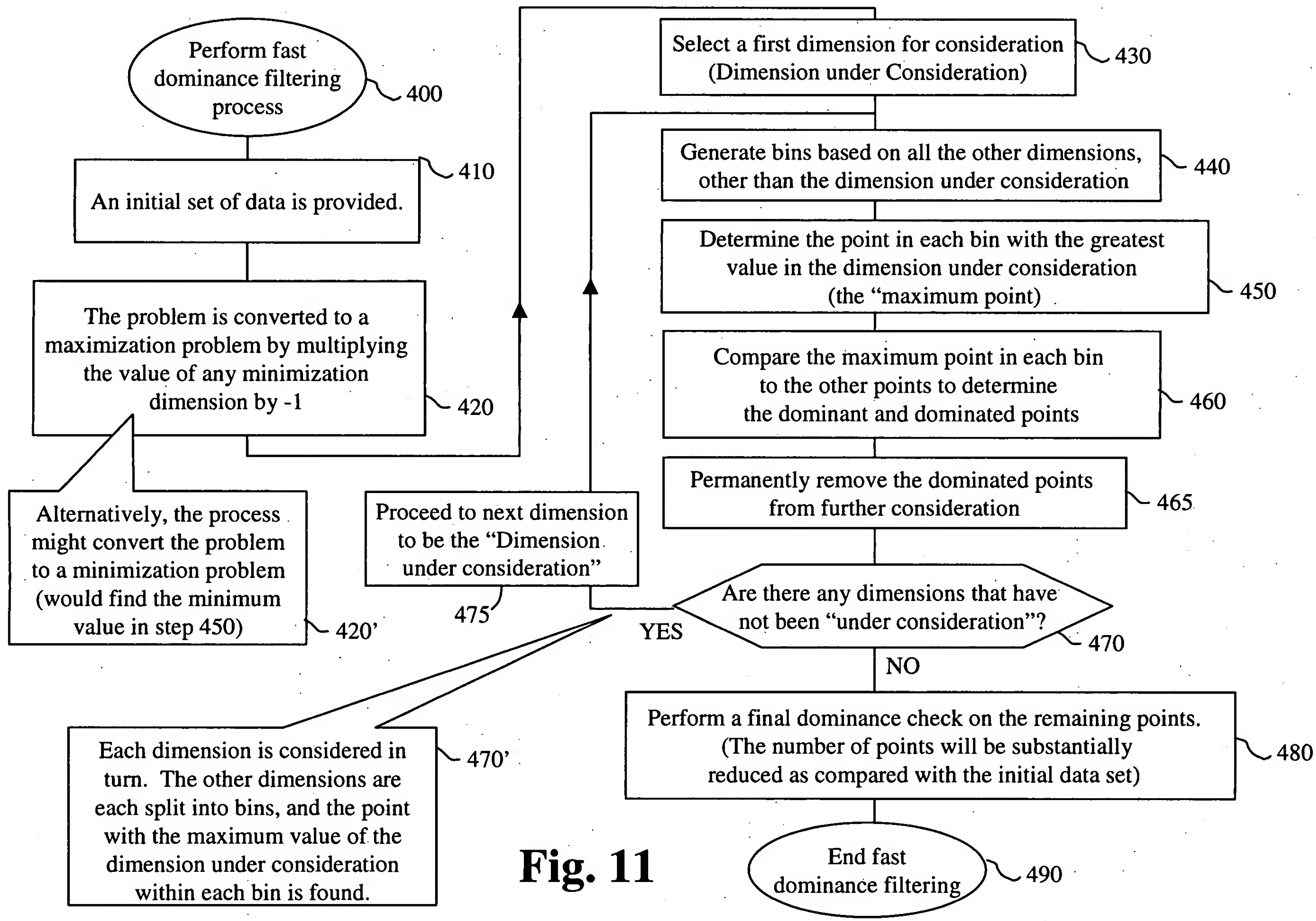

Perform fast dominance filtering process
400
An initial set of data is provided.
410
The problem is converted to a maximization problem by multiplying the value of any minimization dimension by -1
420
Alternatively, the process might convert the problem to a minimization problem (would find the minimum value in step 450)
420'
Select a first dimension for consideration (Dimension under Consideration)
430
Generate bins based on all the other dimensions, other than the dimension under consideration
440
Determine the point in each bin with the greatest value in the dimension under consideration (the "maximum point)
450
Compare the maximum point in each bin to the other points to determine the dominant and dominated points
460
Permanently remove the dominated points from further consideration
465
Are there any dimensions that have not been "under consideration"?
470
YES
NO
Proceed to next dimension to be the "Dimension under consideration"
475
Each dimension is considered in turn. The other dimensions are each split into bins, and the point with the maximum value of the dimension under consideration within each bin is found.
470'
Perform a final dominance check on the remaining points. (The number of points will be substantially reduced as compared with the initial data set)
480
End fast dominance filtering
490

Fig. 11

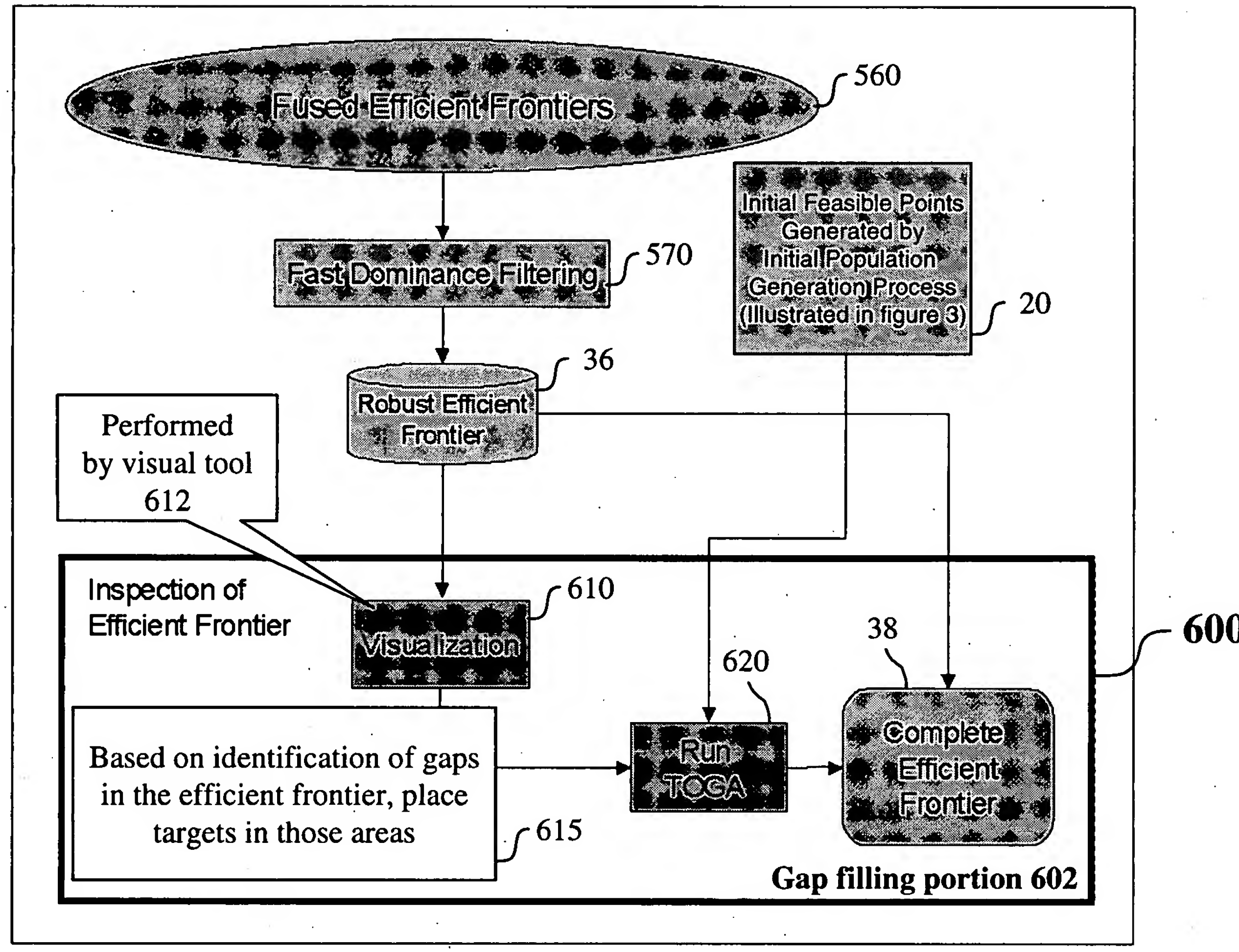
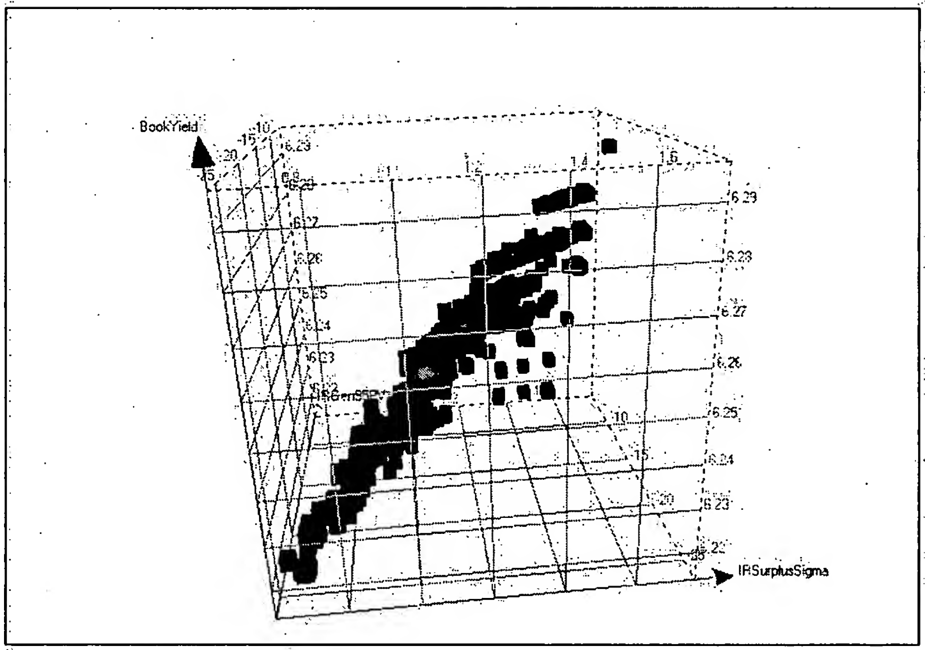


Fig. 12

Process to interactively fill any gaps in the identified efficient frontier

Fig. 13



Efficient Frontier in a 3D View

Example of Parallel coordinate plot

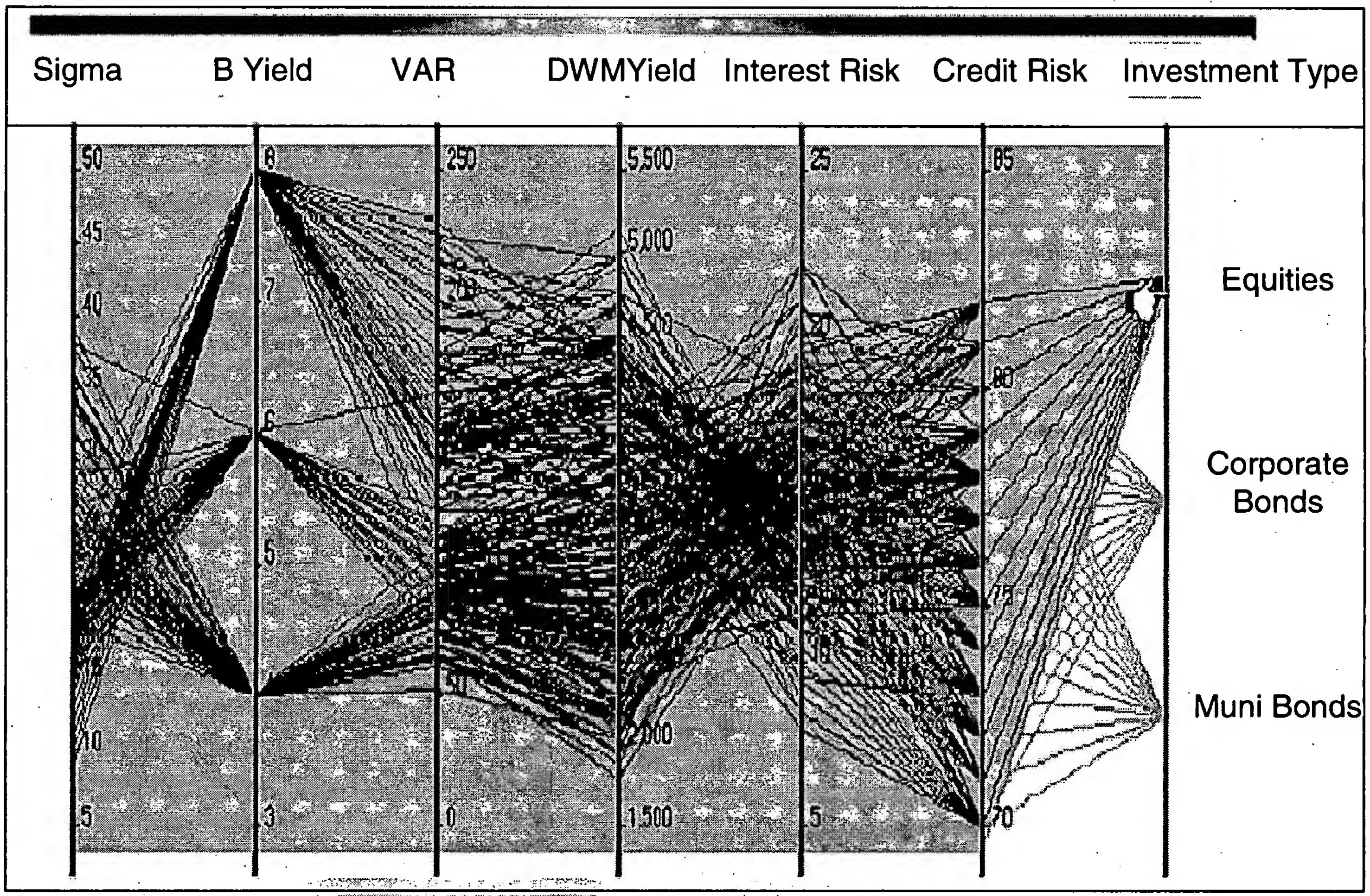


Fig. 14

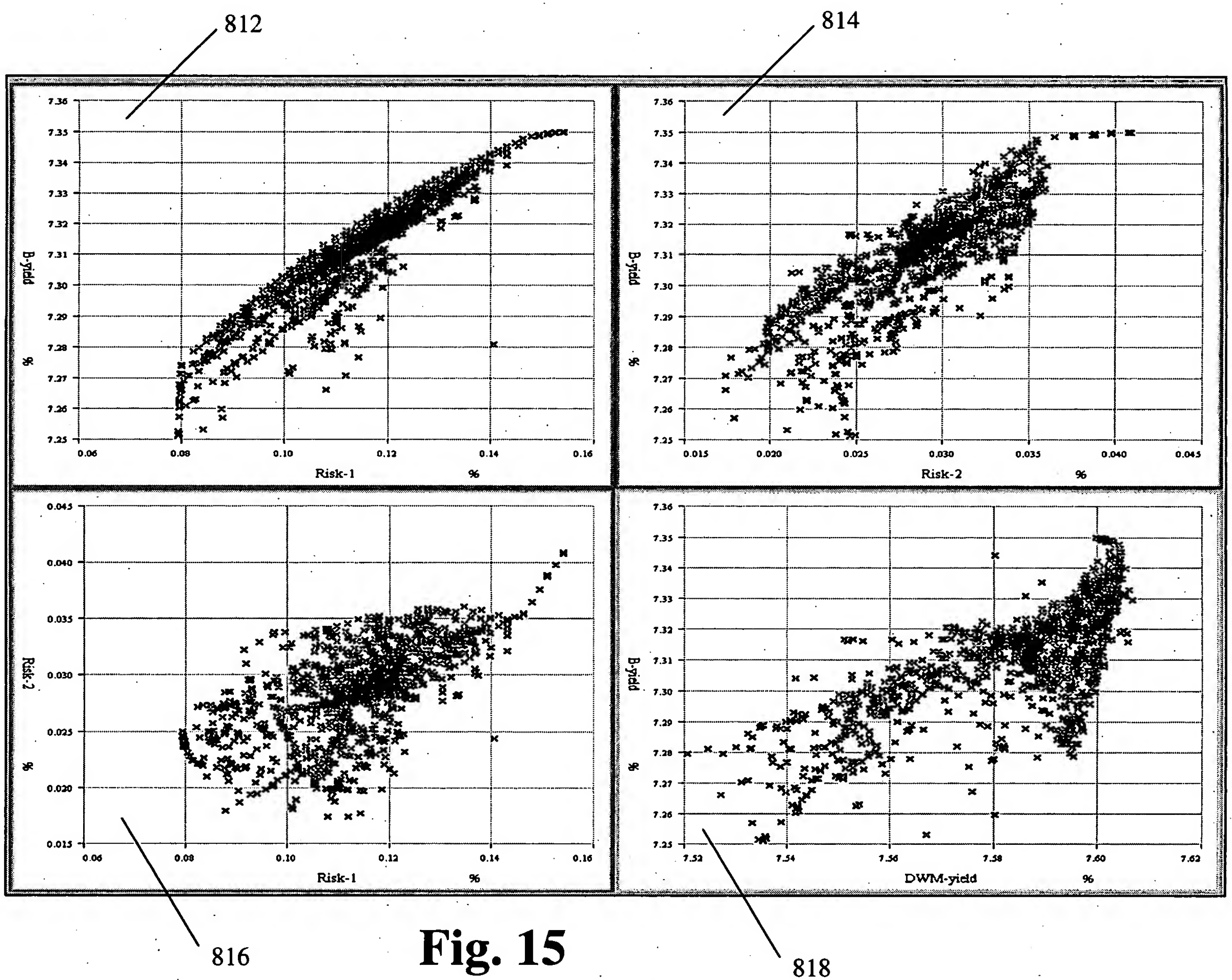


Fig. 15

Fig. 16

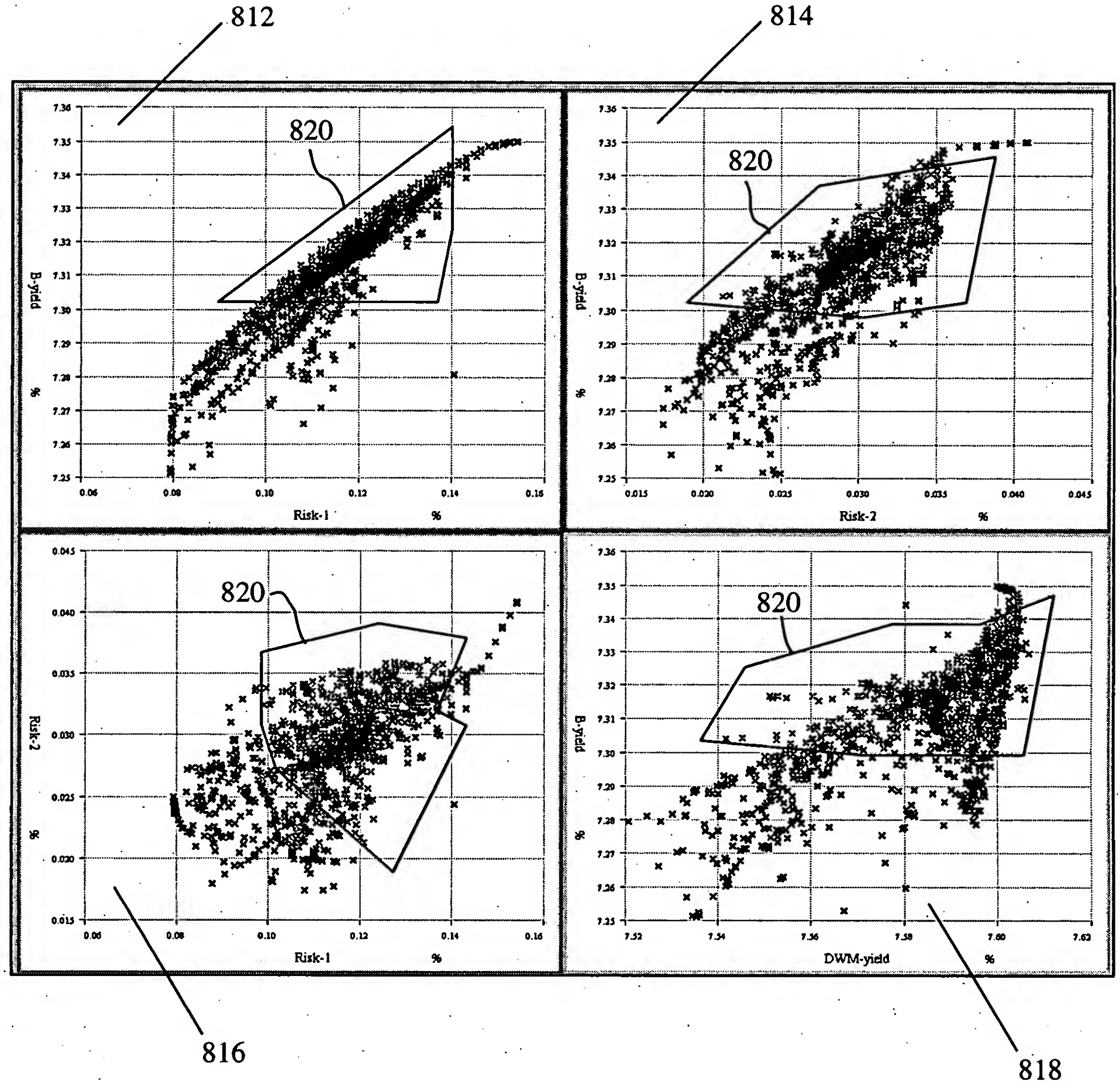

812
814
816
818
820
820
820
820
B-yield %
Risk-1 %
B-yield %
Risk-2 %
Risk-2 %
Risk-1 %
B-yield %
DWM-yield %

Fig. 17

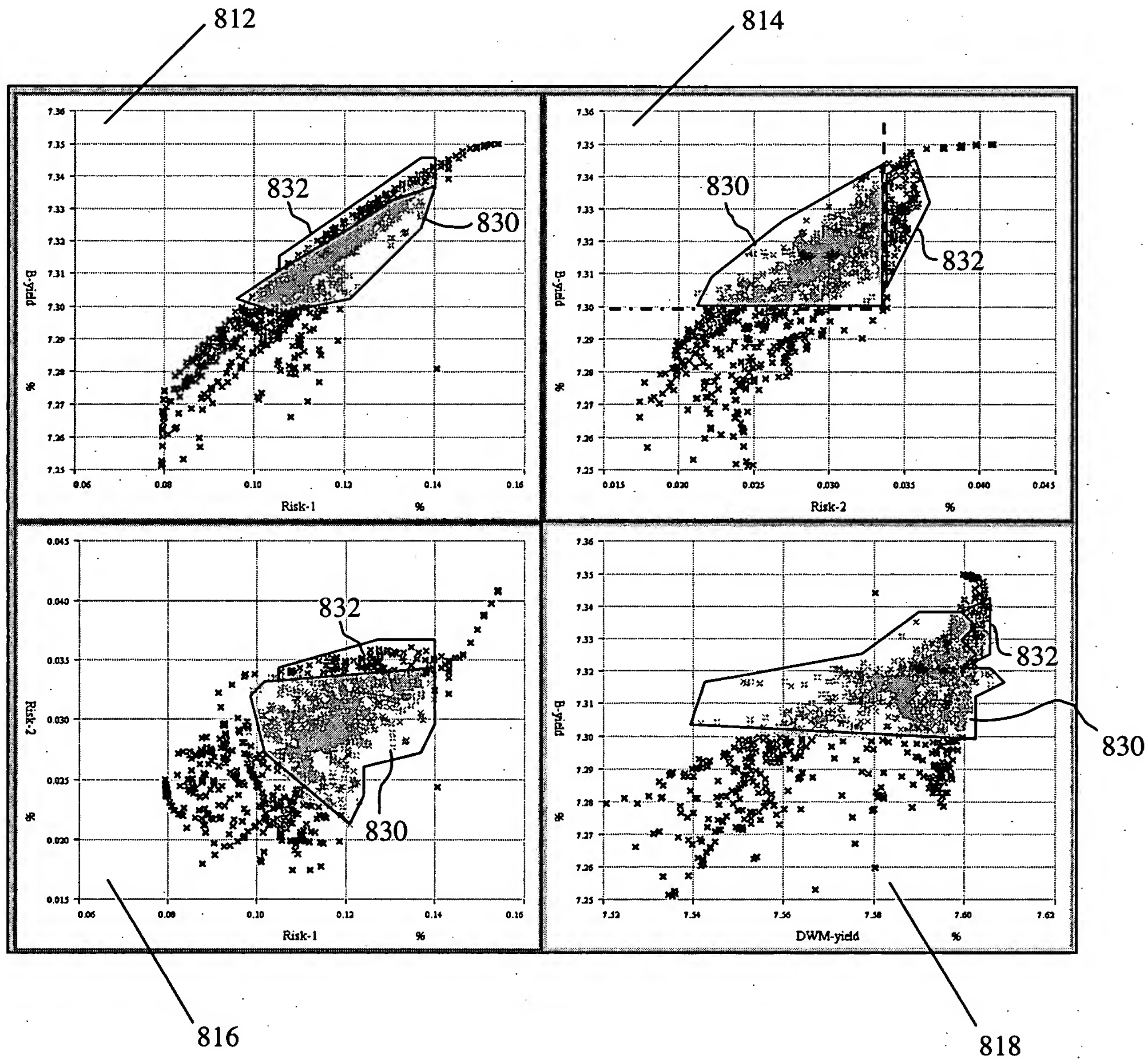
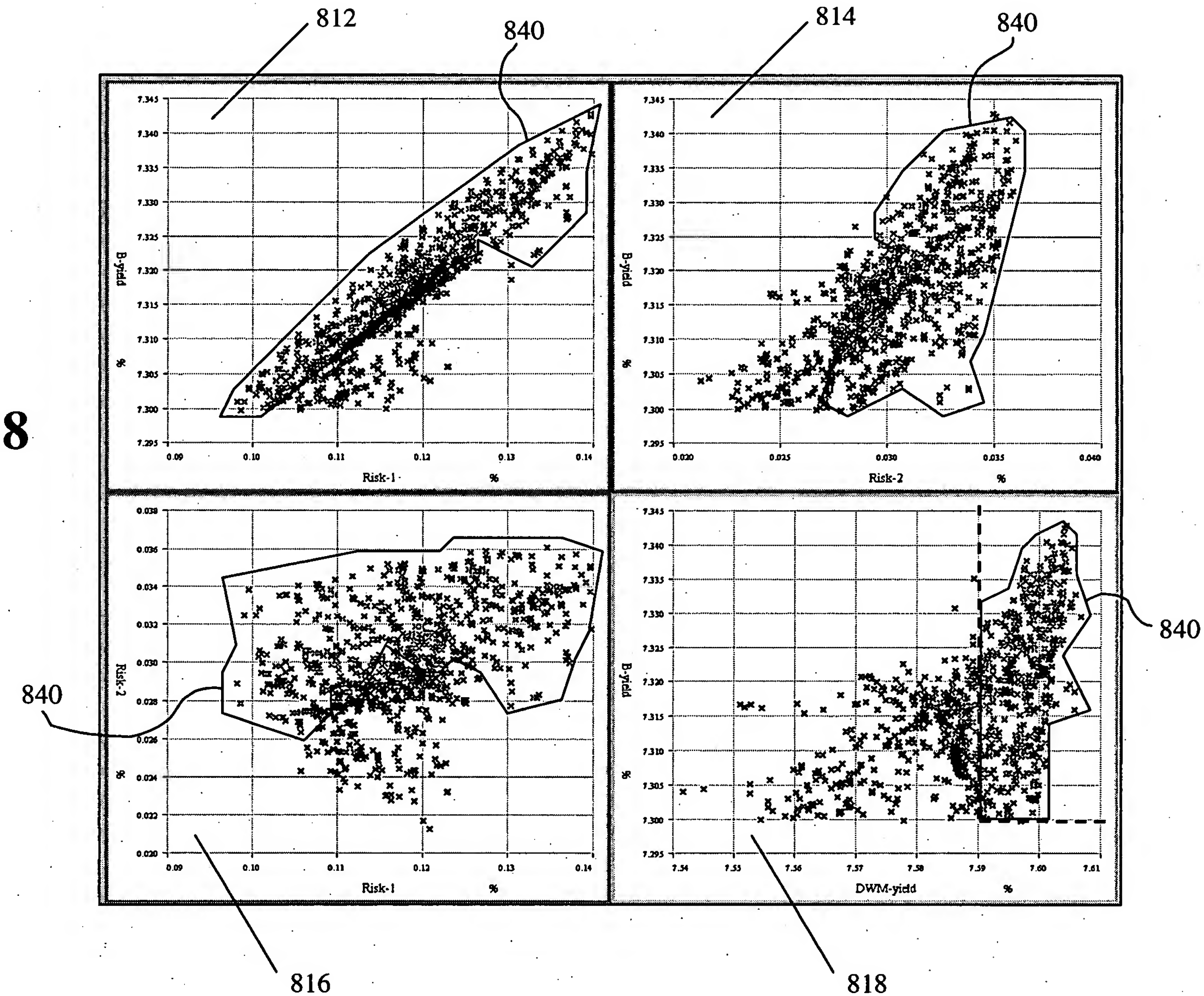


Fig. 18



812
840
814
840
B-yield
%
Risk-1
%
B-yield
%
Risk-2
%
Risk-2
%
Risk-1
%
B-yield
%
DWM-yield
%
840
840
816
818

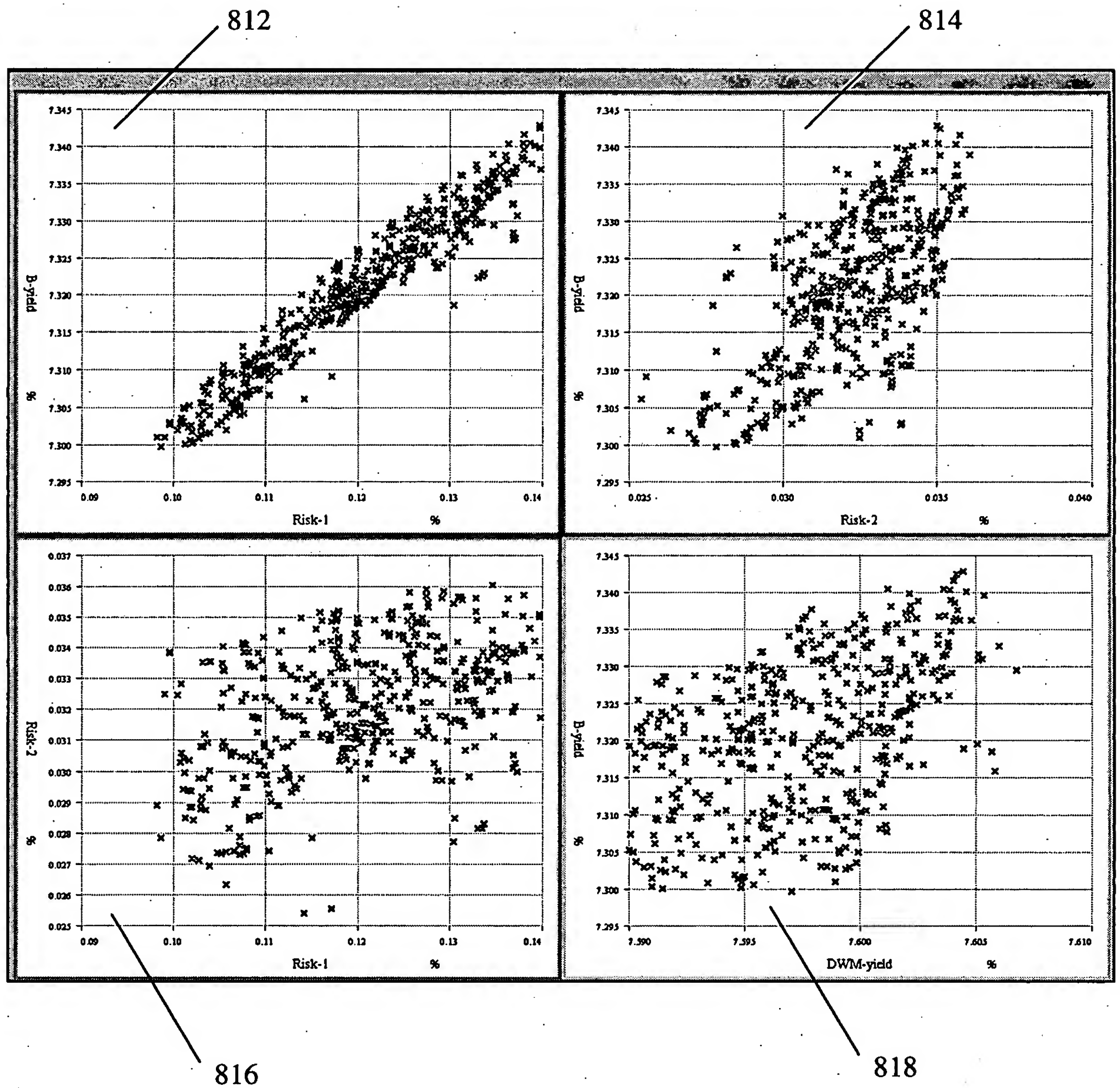


Fig. 19

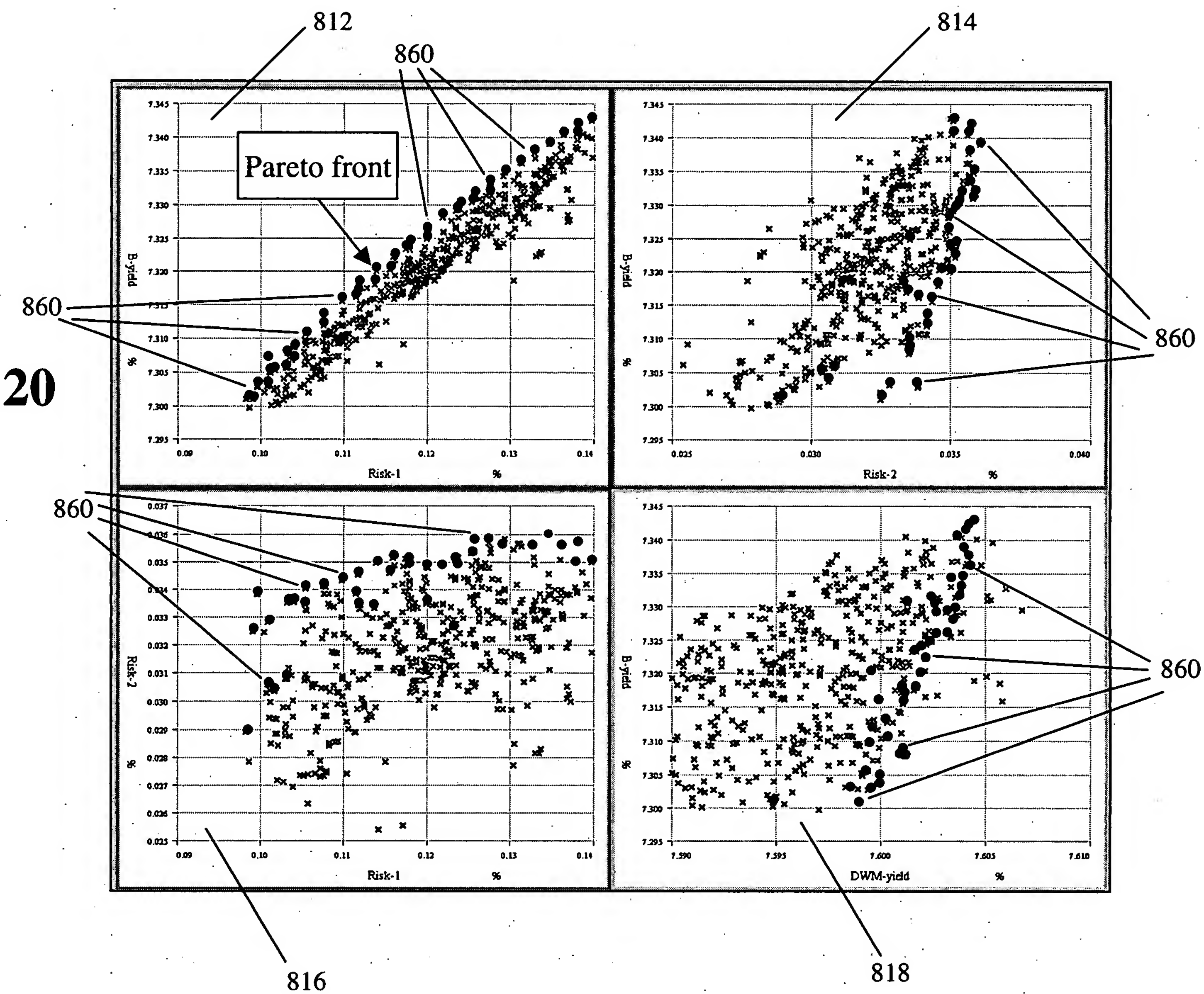
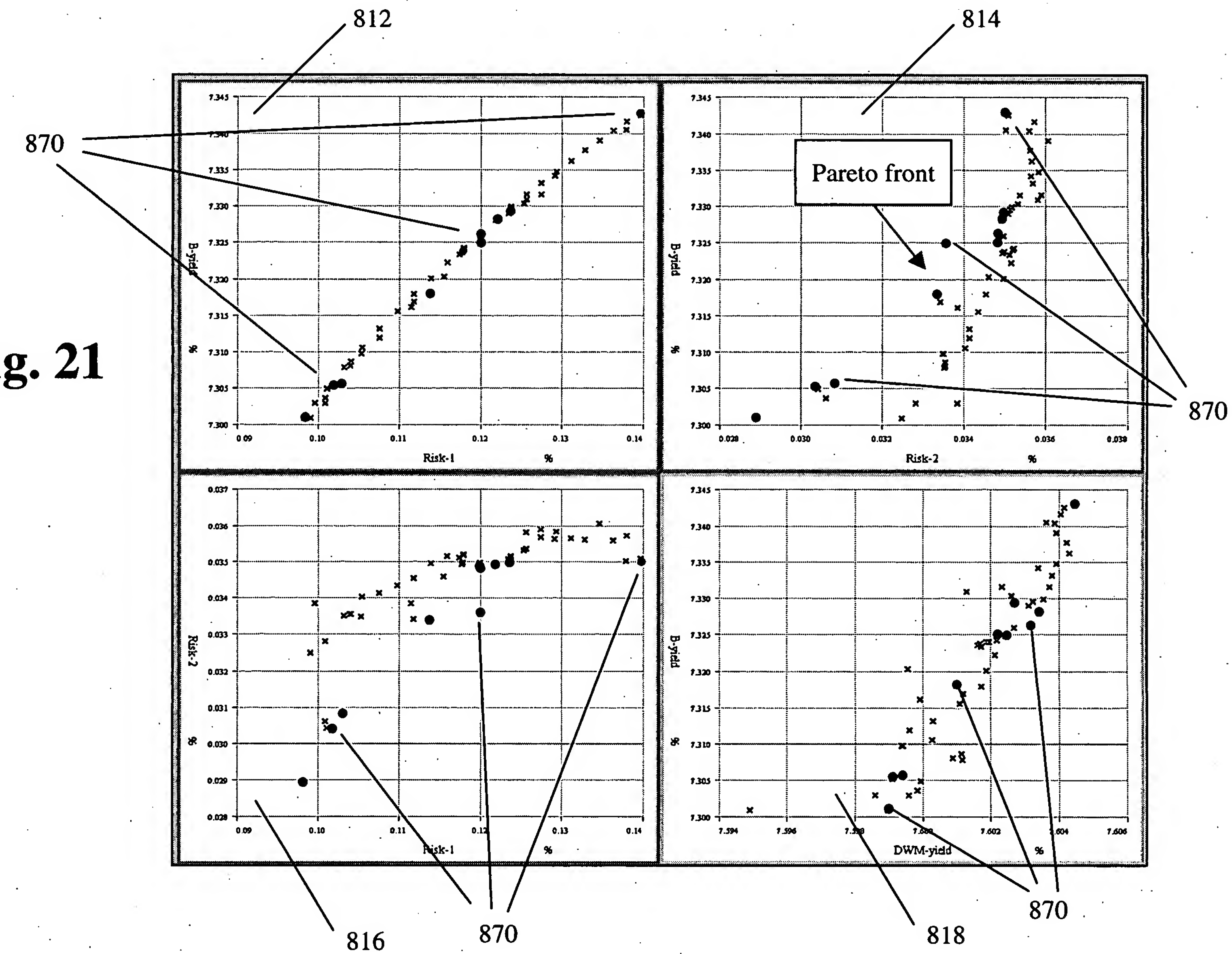

812
814
816
818
860
Pareto front
B-yield
%
Risk-1
Risk-2
DWM-yield

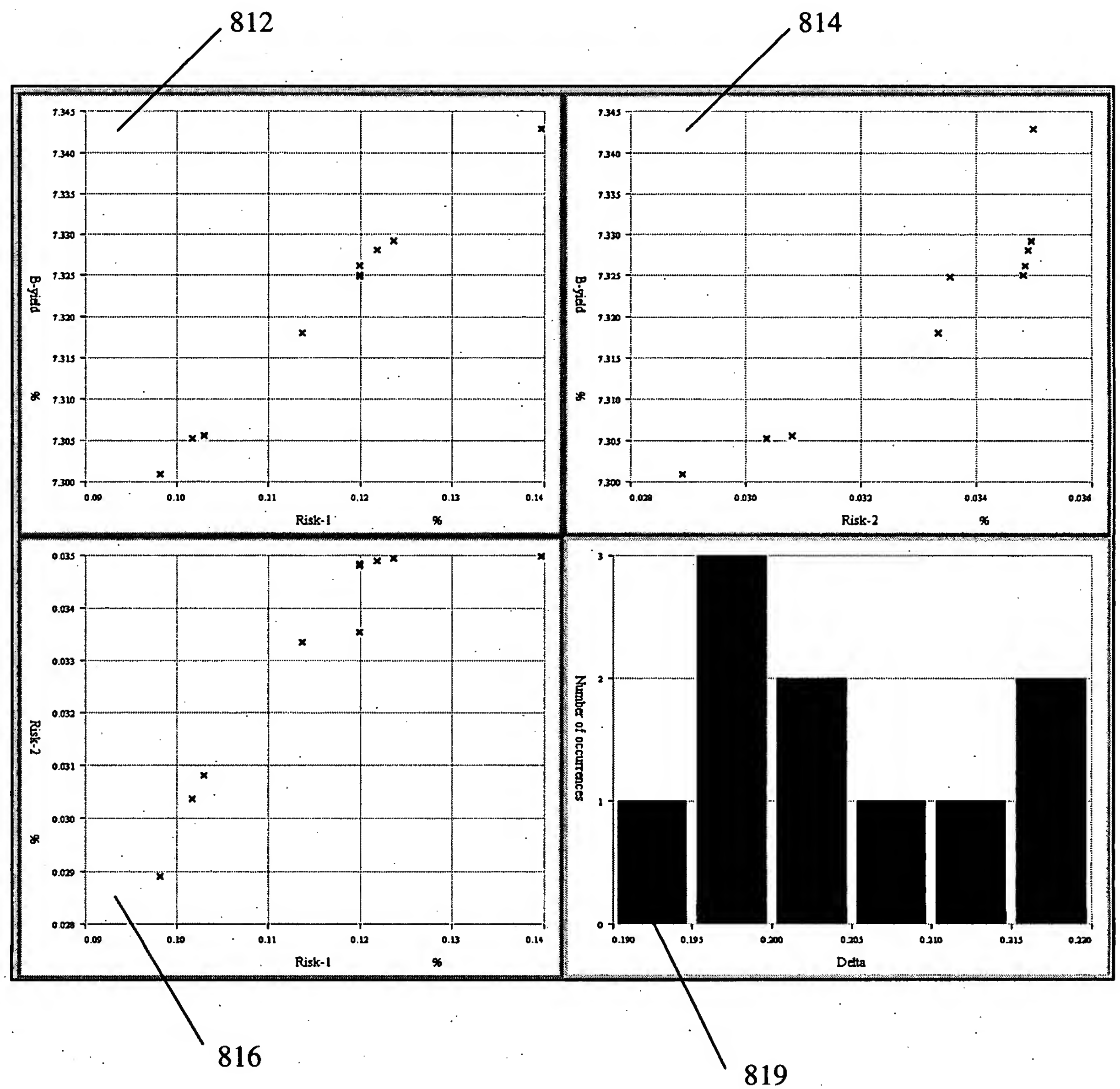
Fig. 20

Fig. 21



812
814
870
Pareto front
B-yield
%
Risk-1
Risk-2
DWM-yield
816
818

Fig. 22


812
814
816
819
B-yield %
Risk-1 %
Risk-2 %
Number of occurrences
Delta
7.345
7.340
7.335
7.330
7.325
7.320
7.315
7.310
7.305
7.300
0.09
0.10
0.11
0.12
0.13
0.14
0.028
0.030
0.032
0.034
0.036
0.035
0.034
0.033
0.032
0.031
0.030
0.029
0.028
0
1
2
3
0.190
0.195
0.200
0.205
0.210
0.215
0.220

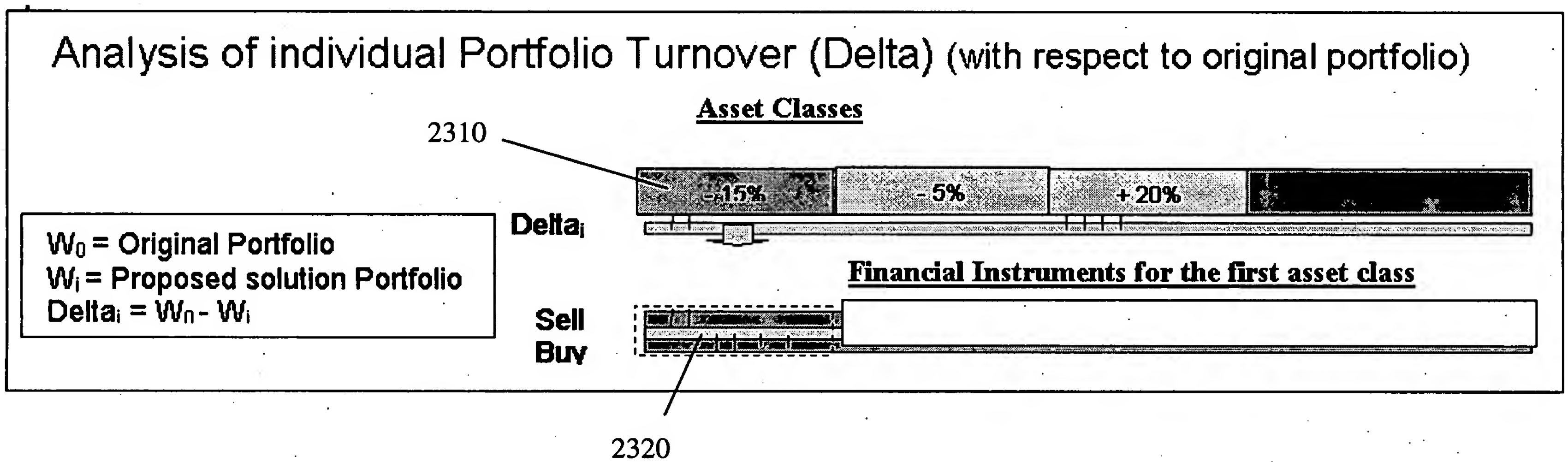

Analysis of individual Portfolio Turnover (Delta) (with respect to original portfolio)
Asset Classes
2310
-15%
-5%
+20%
Delta$_i$
W_0 = Original Portfolio
W_i = Proposed solution Portfolio
Delta$_i$ = W_n - W_i
Financial Instruments for the first asset class
Sell
Buy
2320

Fig. 23

Allocation	Asset Class 1	Asset Class 2	Asset Class 3	Asset Class 4	Asset Class 5	Total
Original Portfolio	35%	20%	5%	15%	25%	100%
P1	20%	15%	25%	15%	25%	100%
P2	40%	25%	10%	10%	15%	100%
P3	20%	20%	15%	20%	25%	100%
P4	15%	30%	20%	20%	15%	100%
P5	45%	20%	15%	10%	10%	100%
P6	20%	25%	20%	25%	10%	100%
P7	25%	25%	15%	20%	15%	100%
P8	30%	15%	10%	25%	20%	100%
P9	20%	25%	15%	20%	20%	100%
P10	30%	10%	15%	25%	20%	100%

Fig. 24

Deltas	Asset Class 1	Asset Class 2	Asset Class 3	Asset Class 4	Asset Class 5	Net Change
P1	-15%	-5%	20%	0%	0%	0%
P2	5%	5%	5%	-5%	-10%	0%
P3	-15%	0%	10%	5%	0%	0%
P4	-20%	10%	15%	5%	-10%	0%
P5	10%	0%	10%	-5%	-15%	0%
P6	-15%	5%	15%	10%	-15%	0%
P7	-10%	5%	10%	5%	-10%	0%
P8	-5%	-5%	5%	10%	-5%	0%
P9	-15%	5%	10%	5%	-5%	0%
P10	-5%	-10%	10%	10%	-5%	0%
Average	**-9%**	**1%**	**11%**	**4%**	**-8%**	
Median	**-13%**	**3%**	**10%**	**5%**	**-8%**	

Fig. 25

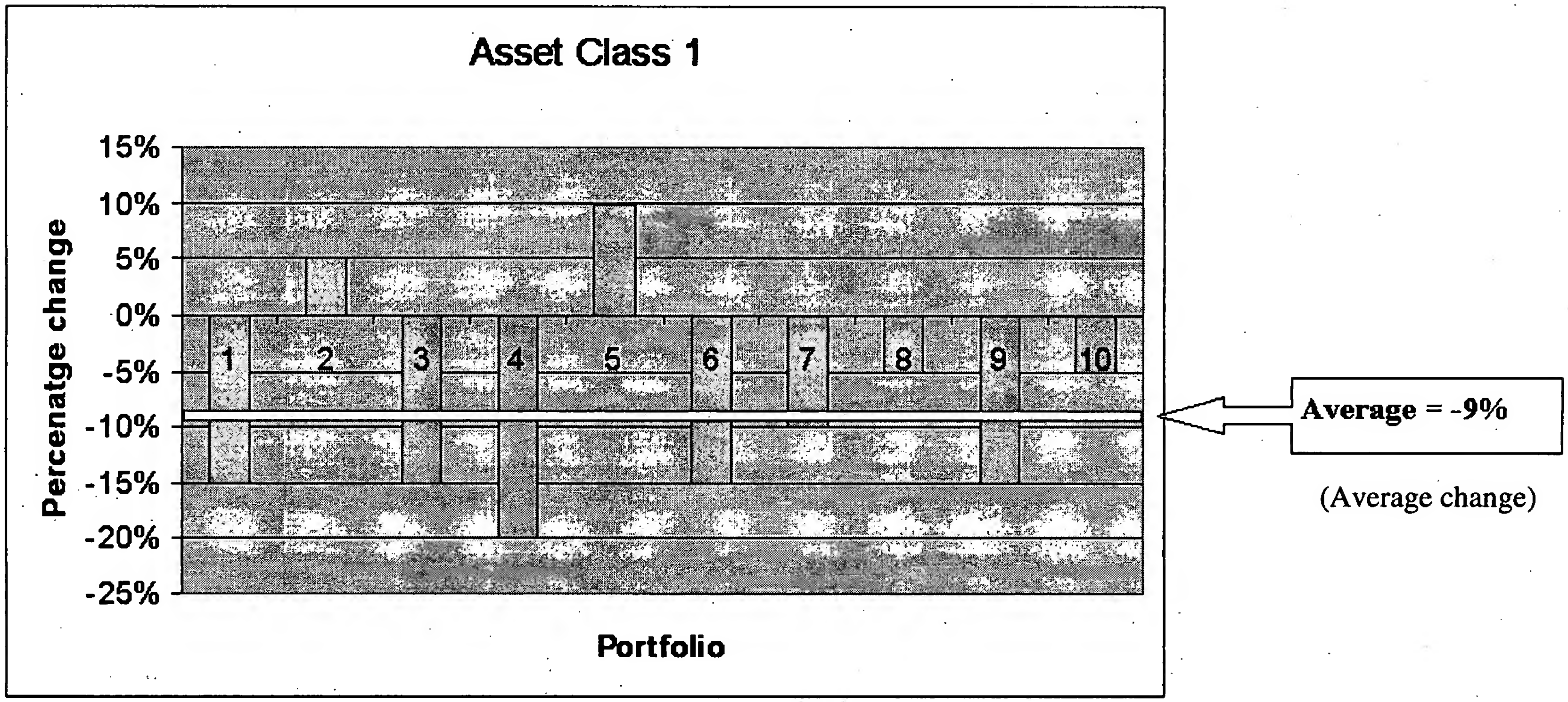

Asset Class 1
15%
10%
5%
0%
-5%
-10%
-15%
-20%
-25%
Percenatge change
1
2
3
4
5
6
7
8
9
10
Portfolio
Average = -9%
(Average change)

Fig. 26

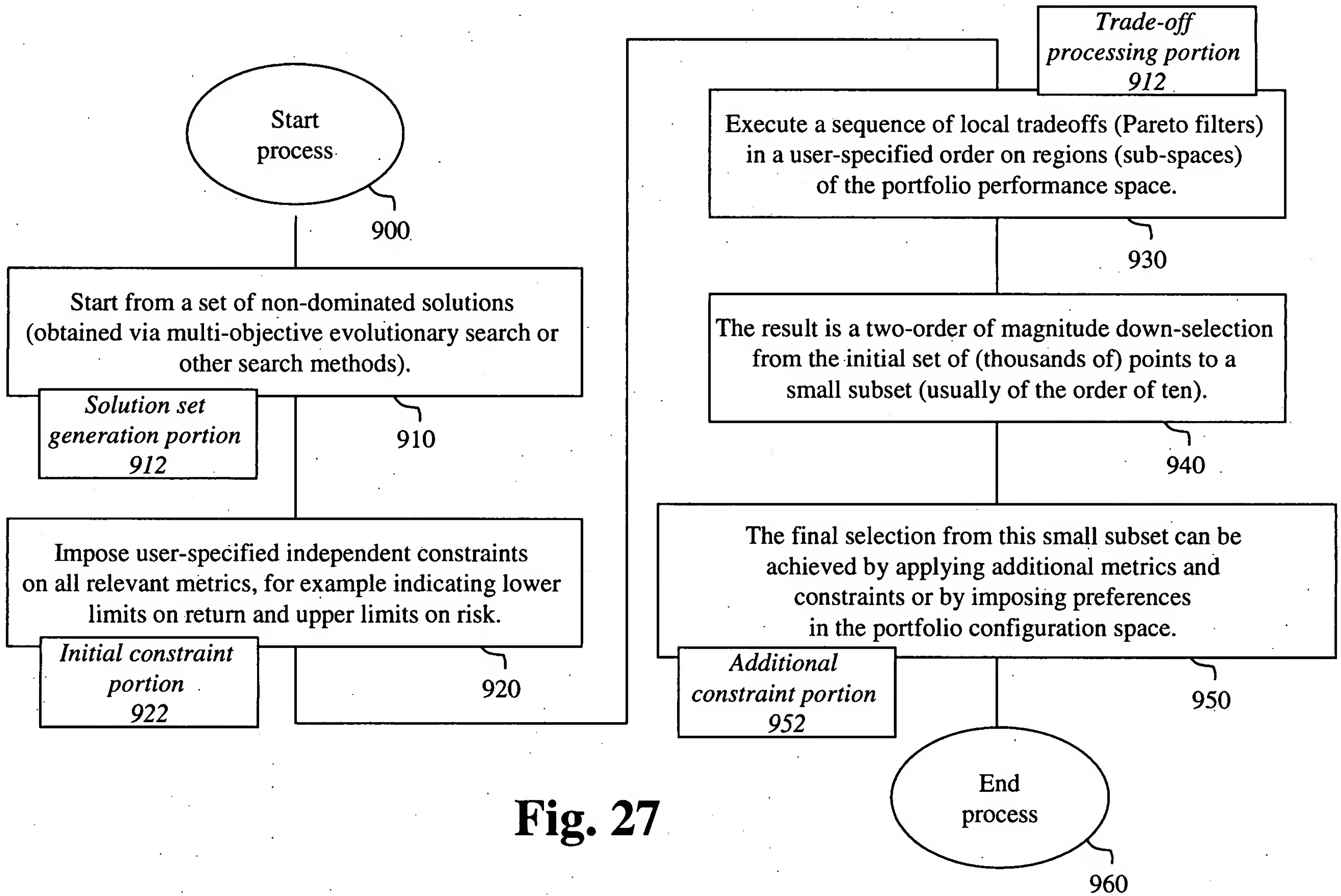

Start process
900
Start from a set of non-dominated solutions (obtained via multi-objective evolutionary search or other search methods).
Solution set generation portion 912
910
Impose user-specified independent constraints on all relevant metrics, for example indicating lower limits on return and upper limits on risk.
Initial constraint portion 922
920
Trade-off processing portion 912
Execute a sequence of local tradeoffs (Pareto filters) in a user-specified order on regions (sub-spaces) of the portfolio performance space.
930
The result is a two-order of magnitude down-selection from the initial set of (thousands of) points to a small subset (usually of the order of ten).
940
The final selection from this small subset can be achieved by applying additional metrics and constraints or by imposing preferences in the portfolio configuration space.
Additional constraint portion 952
950
End process
960

Fig. 27

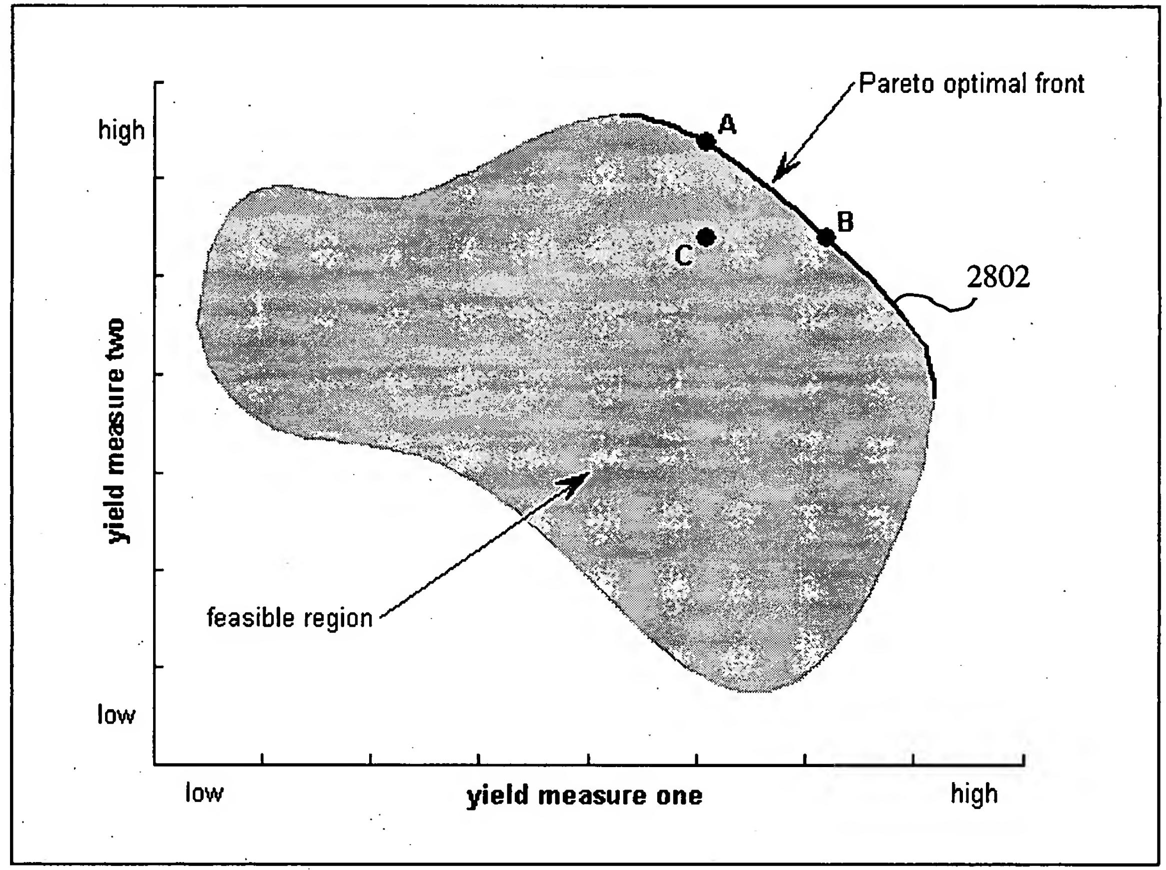
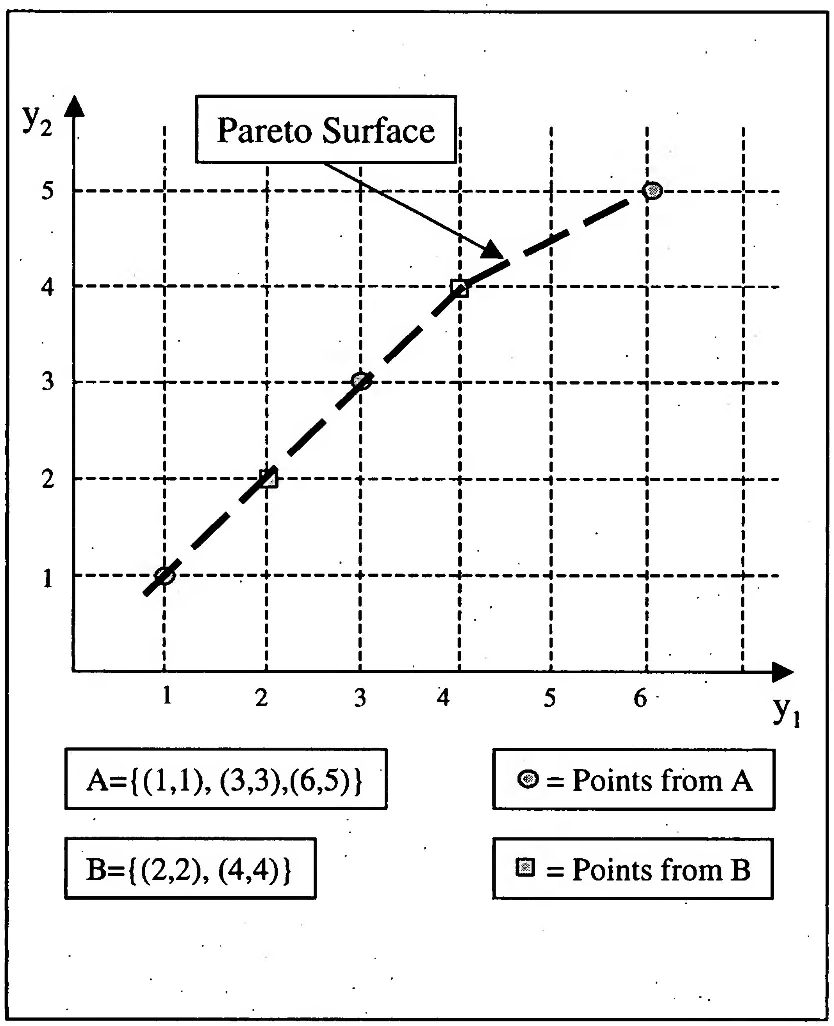

Pareto optimal front
A
B
C
2802
high
yield measure two
low
feasible region
low
yield measure one
high

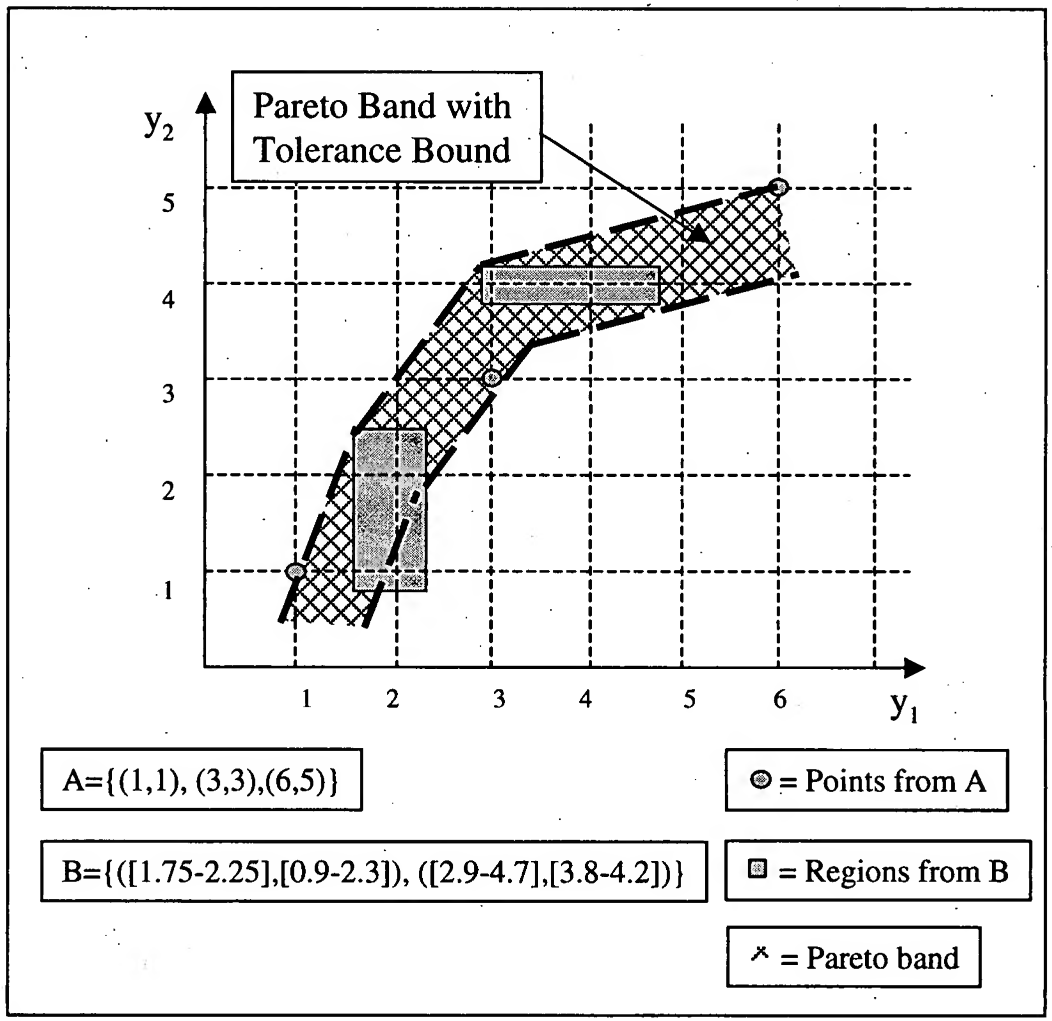
Fig. 28



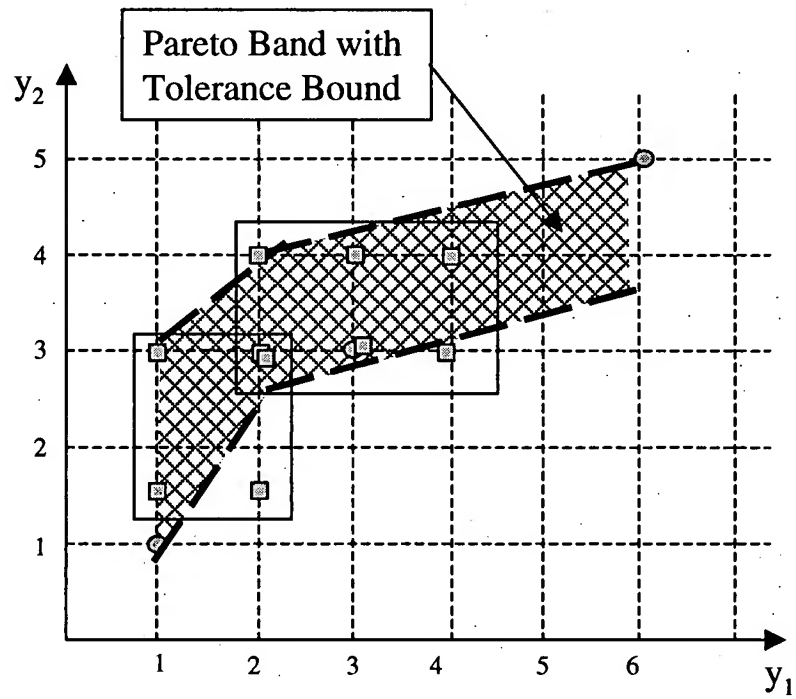
Deterministic Evaluation

Figure 29

Figure 30



Stochastic Evaluation (Transformed into Confidence Intervals)



A ={(1,1), (3,3, (6,5)}

B={([p(1) = m1, (p(2) =1-m1],[(p(1.5)=m3, p(3)=1-m3]),
([p(2) = m4, p(3)= m5, p(4)= 1-m4-m5],[p(3) = m6, (p(4) =1-m6])}

Figure 31

Discrete Probabilistic Evaluation

Figure 32

$$A=\{\quad p_1(1,1)=1,\quad \mathbf{p_2(3,3)=1},\quad p_3(6,5)=1\}$$

$$\begin{aligned}
B=\{\{\ & p_4(1, 1.5) = m1*m3 \\
& p_4(1, 3) = m1*(1-m3) \\
& p_4(2, 1.5) = (1-m1)*m3, \\
& \mathbf{p_4 (2, 3) = (1-m1)*(1-m3)}\}, \\
\\
\{ & \mathbf{p_5(2, 3) = m4*m6} \\
& \mathbf{p_5(3, 3) = m5*m6} \\
& p_5(4, 3) = (1-m4-m5)*m6 \\
& p_5(2, 4) = m4*(1-m6) \\
& p_5(3, 4) = m5 *(1-m6) \\
& p_5(4, 4) = (1-m4-m5 *(1-m6)\}\ \}
\end{aligned}$$

Fusion (PF) of multiple assignments to the same point:

$$\begin{aligned}
PF(2,3 &= p_4(2,3) + p_5(2,3) - p_4(2,3)*p_5(2,3) \\
&= (1-m1)*(1-m3)+ m4*m6 - [(1-m1)*(1-m3)* m4*m6
\end{aligned}$$

$$\begin{aligned}
PF(3,3) &= p_2(3, 3)+ p_5(3, 3) - p_2(3, 3)* p_5(3, 3) \\
&= 1+ m5*m6 - 1* m5*m6 = 1
\end{aligned}$$

Probabilistic Fusion

Feasible Regions for Optimization Figure 33

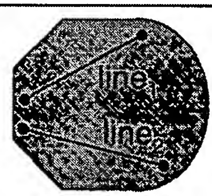
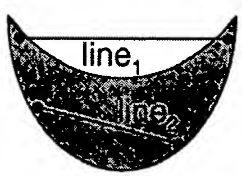
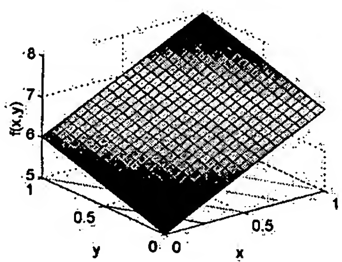
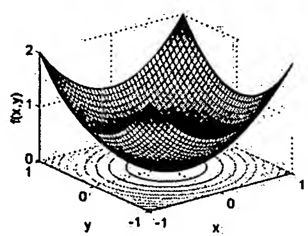
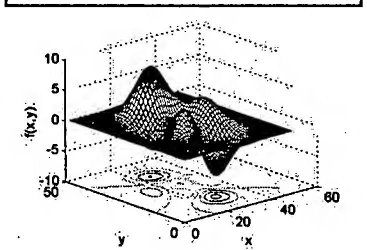
Graphic Visual	Word Description	Example Equation	GEAM
Linear Convex Space 	• For any two points in the space, the line connecting the two points is always contained in the same space • Space is defined using linear equations	$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \vdots & \vdots \\ a_{81} & a_{82} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_8 \end{bmatrix}$ Set of linear equations	• Market value weighted yield formulation • Duration weighted yield formulation
Nonlinear Convex Space 	• For any two points in the space, the line connecting the two points is always contained in the same space • Space is defined using some nonlinear equations	$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ \vdots & \vdots \\ a_{51} & a_{52} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \le \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_5 \end{bmatrix}$ $x^2 + y^2 \le \alpha$ Nonlinear equation	• Interest rate sigma formulation
Nonlinear Nonconvex Space 	• For any two points in the space, the line connecting the two points is <u>not</u> always contained in the same space • Space is defined using some nonlinear equations	$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \begin{bmatrix} x^2 \\ x \\ y \end{bmatrix} \le \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$ Set of nonlinear equations	• Interest rate sigma and VAR formulation • VAR is a nonlinear nonconvex constraint

Figure 34

Objective Functions

Graphic Visual	Word Description	Example Equation	GEAM
Linear Function 	• Function is defined using ***linear*** equations • Straightforward math relationship • ***Easy to optimize***	$f(x, y) = 2x + y + 5$	• Market value weighted yield • Duration weighted yield
Nonlinear Convex Function 	• Function is defined using a ***nonlinear*** equation • Functional gradients lead to single optimum • ***Harder to optimize***	$f(x, y) = x^2 + y^2$	• Interest rate sigma
Nonlinear Nonconvex Function 	• Function is defined using ***complex nonlinear*** equations • Multiple local optima • Functional gradients are inefficient • ***Very hard to optimize***	$f(x,y) = g_1(x,y) + g_2(x,y) + g_3(x,y) + g_4(x,y)$	• Interest rate sigma and VAR

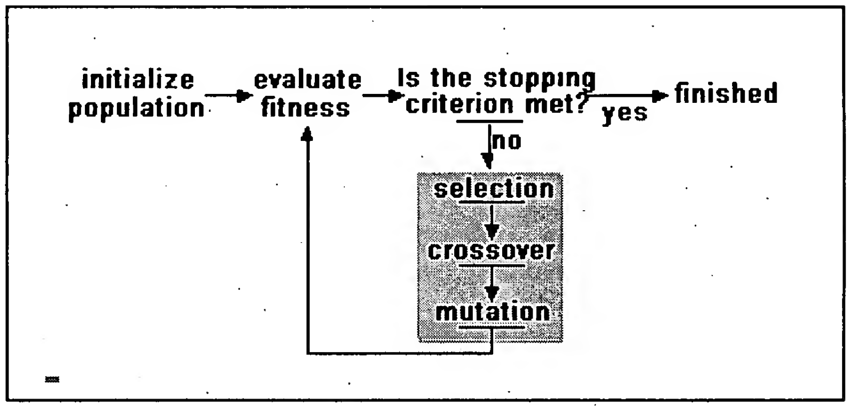
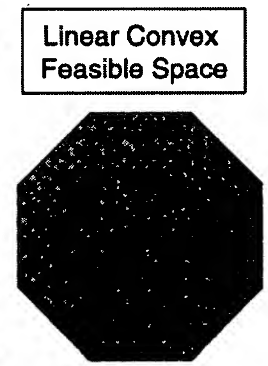

initialize population
evaluate fitness
Is the stopping criterion met?
yes
finished
no
selection
crossover
mutation

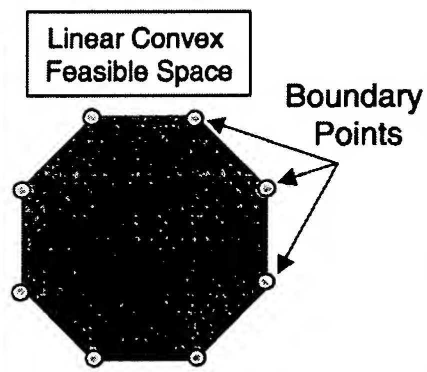
Figure 35

Evolutionary Search Augmented with Domain Knowledge

Multi-objective portfolio optimization problem is formulated as a problem with Multiple linear, nonlinear and nonlinear nonconvex objectives. However, the **domain knowledge** allows us to use strictly **linear and convex constraints.**



Knowledge about geometry of feasible space (i.e. convexity), allowed us develop a feasible space boundary sampling algorithm (**solutions archive generation**). By knowing the boundary of the search space, we can exploit that knowledge to design efficient interior sampling methods.



Convex crossover is a powerful interior sampling method, which is guaranteed to produce feasible offspring solutions. Given parents P_1, P_2, it creates offspring $O_1 = \lambda P_1 + (1-\lambda)P_2$, $O_2 = (1-\lambda)P_1 + \lambda P_2$. An offspring O_k and P_k can crossed over to produce more **diverse** offspring.

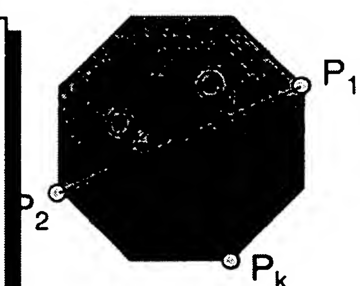


Figure 36

Example of Outer Product using as operator the function $T(x,y)$

T-norm	**Correlation Type**
$T_1(x,y) = \max(0, x+y-1)$	Extreme case of negative correlation
$T_2 = x * y$	No correlation
$T_3 = \min(x,y)$	Extreme case of positive correlation

Figure
37

Example of Outer Product using as operator the function $S(x,y)$

T-conorm	**Correlation Type**
$S_1 = \min(1, x + y)$	Extreme case of negative correlation
$S_2 = x + y - (x * y)$	No correlation
$S_3 = \max(x, y)$	Extreme case of positive correlation

Figure
38